Joint Estimation of Musical Content Information From an Audio Signal

Hélène Papadopoulos

THESE DE DOCTORAT DE
L'UNIVERSITE PIERRE ET MARIE CURIE

Spécialité

informatique, télécommunications et électronique
(Ecole doctorale Edite)

Présentée par

Hélène PAPADOPOULOS

Pour obtenir le grade de

DOCTEUR de l'UNIVERSITÉ PIERRE ET MARIE CURIE

Sujet de la thèse :

Estimation conjointe d'information de contenu musical d'un signal audio

Soutenue le 2 juillet 2010 devant le jury composé de :

M. Gaël RICHARD (Professeur)	Président du jury
M. Xavier RODET (Professeur).	Directeur de thèse
M. Frédéric BIMBOT (Directeur de recherches, HDR) M. Mark SANDLER (Professeur)	Rapporteurs
Mme. Régine ANDRÉ-OBRECHT (Professeur) M. Geoffroy PEETERS (Chercheur) M. Dominique POLACK (Professeur) M. Xavier SERRA (Professeur)	Examinateurs

Université Pierre & Marie Curie - Paris 6
Bureau d'accueil, inscription des doctorants et base de données
Esc G, 2ème étage
15 rue de l'école de médecine
75270-PARIS CEDEX 06

Tél. Secrétariat : 01 42 34 68 35
Fax : 01 42 34 68 40
Tél. pour les étudiants de A à EL : 01 42 34 69 54
Tél. pour les étudiants de EM à MON : 01 42 34 68 41
Tél. pour les étudiants de MOO à Z : 01 42 34 68 51
E-mail : scolarite.doctorat@upmc.fr

2010-07-02
Ph.D. Thesis

JOINT ESTIMATION OF MUSICAL CONTENT INFORMATION FROM AN AUDIO SIGNAL

Hélène Papadopoulos

IRCAM
1, Place Igor Stravinsky
75005 Paris, FRANCE

Remerciements

Je remercie M. Xavier Rodet de m'avoir accordé sa confiance en m'accueillant au sein de l'équipe *Analyse/Synthèse* qu'il dirige à l'IRCAM et de m'avoir permis de travailler sur un sujet très stimulant d'un point de vue scientifique et musical.

Je remercie M. Geoffroy Peeters pour avoir su partager son dynamisme et son excellence scientifique avec une grande attention, rendant nos discussions toujours enrichissantes.

Mes travaux de recherche ont été financés par les projets *Écoute* et *QUAERO*. Participer à l'avancement de ces projets et aux discussions scientifiques associées a été une grande source de motivations.

Au cours de mes travaux de thèse, j'ai collaboré à certaines occasions avec d'autres scientifiques que je voudrais remercier ici : M. Ichiro Fujinaga (université de McGill, Montréal, Canada) qui m'a accueillie à deux reprises dans son équipe ; M. Matthew Davies (université Queen Mary, Londres) qui m'a invitée à donner une conférence et a mis à ma disposition son code de calcul de la position des premiers temps ; M. Frédéric Bimbot (INRIA, Rennes), qui m'a donné l'opportunité de présenter mes travaux et de rencontrer les chercheurs de son équipe ; M. Chungsin Yeh (IRCAM) qui m'a permis d'utiliser le code sur l'estimation des fréquences fondamentales multiples qu'il a développé.

Enfin, je remercie l'ensemble du personnel de l'IRCAM, et plus particulièrement les membres de l'équipe *Analyse/Synthèse*, pour avoir créé un environnement de travail agréable et motivant.

Résumé

Dans cette thèse, nous nous intéressons au problème de l'extraction automatique d'informations de contenu d'un signal audio de musique. La plupart des travaux existants abordent ce problème en considérant les attributs musicaux de manière indépendante les uns vis-à-vis des autres. Cependant les morceaux de musique sont extrèmement structurés du point de vue de l'harmonie et du rythme et leur estimation devrait se faire en tenant compte du contexte musical, comme le fait un musicien lorsqu'il analyse un morceau de musique.

Nous nous concentrons sur trois descripteurs musicaux liés aux structures harmoniques, métriques et tonales d'un morceau de musique. Plus précisément, nous cherchons à en estimer la progression des accords, les premiers temps et la tonalité. L'originalité de notre travail consiste à construire un modèle qui permet d'estimer de manière conjointe ces trois attributs musicaux. Notre objectif est de montrer que l'estimation des divers descripteurs musicaux est meilleure si on tient compte de leurs dépendances mutuelles que si on les estime de manière indépendante. Nous proposons au cours de ce travail un ensemble de protocoles de comparaison, de métriques de performances et de bases de données de test afin de pouvoir évaluer les différentes méthodes étudiées. Afin de valider notre approche, nous présentons également les résultats de nos participations à des campagnes d'évaluation internationales.

Dans un premier temps, nous examinons plusieurs représentations typiques du signal audio afin de choisir celle qui est la plus appropriée à l'analyse du contenu harmonique d'un morceau de musique. Nous explorons plusieurs méthodes qui permettent d'extraire un *chromagram* du signal et les comparons à travers un protocole d'évaluation original et une nouvelle base de données que nous avons annotée. Nous détaillons et expliquons les raisons qui nous ont amenés à choisir la représentation que nous utilisons dans notre modèle.

Dans notre modèle, les accords sont considérés comme un attribut central autour duquel les autres descripteurs musicaux s'organisent. Nous étudions le problème de l'estimation automatique de la suite des accords d'un morceau de musique audio en utilisant les *chromas* comme observations du signal. Nous proposons plusieurs méthodes basées sur les modèles de Markov cachés (hidden Markov models, HMM), qui permettent de prendre en compte des éléments de la théorie musicale, le résultat d'expériences cognitives sur la perception de la tonalité et l'effet des harmoniques des notes de musique. Les différentes méthodes sont évaluées et comparées pour la première fois sur une grande base de données composée de morceaux de musique populaire.

Nous présentons ensuite une nouvelle approche qui permet d'estimer de manière simultanée la progression des accords et les premiers temps d'un signal audio de musique. Pour cela, nous proposons une topologie spécifique de HMM qui nous permet de modéliser

la dépendance des accords par rapport à la structure métrique d'un morceau. Une importante contribution est que notre modèle peut être utilisé pour des structures métriques complexes présentant par exemple l'insertion ou l'omission d'un temps, ou des changements dans la signature rythmique. Le modèle proposé est évalué sur un grand nombre de morceaux de musique populaire qui présentent des structures métriques variées. Nous comparons les résultats d'un modèle semi-automatique, dans lequel nous utilisons les positions des temps annotées manuellement, avec ceux obtenus par un modèle entièrement automatique où la position des temps est estimée directement à partir du signal.

Enfin, nous nous penchons sur la question de la tonalité. Nous commençons par nous intéresser au problème de l'estimation de la tonalité principale d'un signal audio de musique. Nous étendons le modèle présenté ci-dessus à un modèle qui permet d'estimer simultanément la progression des accords, les premiers temps et la tonalité principale. Les performances du modèle sont évaluées à travers des exemples choisis dans la musique populaire. Nous nous tournons ensuite vers le problème plus complexe de l'estimation de la tonalité locale d'un morceau de musique. Nous proposons d'aborder ce problème en combinant et en étendant plusieurs approches existantes pour l'estimation de la tonalité principale. La spécificité de notre approche est que nous considérons la dépendance de la tonalité locale par rapport aux structures harmoniques et métriques. Nous évaluons les résultats de notre modèle sur une base de données originale composée de morceaux de musique classique que nous avons annotés.

L'estimation automatique des informations de contenu d'un signal audio de musique est un problème complexe. Nous espérons que ce travail est un pas en avant dans cette direction, et qu'il ouvre de nouvelles perspectives.

Abstract

This thesis is concerned with the problem of automatically extracting meaningful content information from music audio signals. Most of the previous works that address the problem of estimating musical attributes from the audio signal have dealt with these elements independently. However, musical elements are deeply related to each other and should be analyzed considering the global musical context, as a musician does when he or she analyzes a piece of music.

Our research concentrates on three musical descriptors related to the harmonic, the metrical and the tonal structure. More specifically, we focus on three musical attributes: the chord progression, the downbeats and the musical key. The scope of this work is to develop a model that allows the joint estimation of the chords, the keys and the downbeats from polyphonic music recordings. We intend to show that integrating knowledge of mutual dependencies between several descriptors of musical content improves their estimation. In our model, harmony is a core around which other musical attributes are organized.

We start by investigating several typical representations of the audio signal in order to select the most appropriate one for the task of harmonic content analysis. We explore several schemes for chromagram computation and investigate several issues related to the use of each representation. We detail and explain the choice of the audio signal representation we use as an input to our model.

We then concentrates on the problem of the automatic estimation of the chord progression, using chroma features as observation of the music signal. From the audio signal, a set of chroma vectors representing the pitch content of the file over time is extracted. The chord progression is then estimated from these observations using a hidden Markov model. Several methods are proposed that allow taking into account music theory, perception of key and presence of higher harmonics of pitch notes. They are evaluated and compared to existing algorithms through a large-scale evaluation on popular music songs.

We then present a new technique for estimating simultaneously the chord progression and the downbeats from an audio file. A specific topology of hidden Markov models that enables modeling chord dependency on the metrical structure is proposed. This model allows us to consider pieces with complex metrical structures such as beat insertion, beat deletion or changes in the meter. The model is evaluated on a large set of popular music songs that present various metrical structures. We compare a semi-automatic model, in which the beat positions are annotated, with a fully automatic model in which a beat tracker is used as a front-end of the system.

Finally, we focus on the problem of key estimation. In a first part, we concentrate on the problem of estimating the main key of a piece. Relying on previous works on key estimation, we extend the above-mentioned model to a model for simultaneous downbeat, chord and key estimation from an audio signal. The model is evaluated on a set of popular music pieces. We then draw our attention to local key finding. We propose to address this problem by investigating the possible combination and extension of different

previous proposed global key estimation approaches. The specificity of our approach is that we introduce key dependency on both the harmonic and the metrical structures. We evaluate and analyze the results of our model on a new annotated database composed of classical music pieces.

Building models for musical content estimation in which the interaction between musical attributes is encoded at the level musicians and trained human listeners do, when they analyze a piece of music, is a very complex problem and one which is far from being solved. However, we hope that our work is a step towards this direction.

Contents

Chapter 1

Introduction

Contents

1.1 Motivations

Within the last few years, the huge explosion of online music collections has become a great source of attention. Specific demands, such as asking an online store to find a song that fits his or her taste and musical expectation among millions of other tracks, became common requirements to music listeners. In this context, techniques for interacting with enormous digital music libraries at the song level are necessary. Content-based music retrieval is therefore a very active and important field of research.

A piece of music can be characterized by a number of musical attributes such as the melody, the chord progression, the instrumentation or the tempo. One of the most important aspects of Music Information Retrieval (MIR) is the extraction and processing of meaningful descriptors from the audio signal. This can be viewed as a subtask of the more general task that is music transcription.

Manual annotation of the content of musical pieces is a very difficult and tedious task that requires a huge amount of effort. It is thus essential to develop techniques for automatically extracting musical elements from musical signals.

This is why there has been an increasing research interest within the last ten years in using computers to analyze music as human beings can do. Humans are able to understand music at different degrees, depending on their level of music training. Because we are immersed with music, music understanding has become an inherent quality of human beings.

Musicians or even non-trained persons are usually able to extract meaningful information when listening to a piece of music. Some tasks, such as following the beats in a music recording, are in general trivial, even for non-musicians, and do not require any particular training.

More complex tasks need some musical training. For instance, identifying the key of a music excerpt or describing music in terms of tonal and harmonic progression requires some theoretical music knowledge. A person without a musical education is usually not able to transcribe chords by ear from a recording whereas trained musicians can accurately label chords from complex polyphonic recordings. This is a common exercise in music academies. Even a non-trained musician can at least feel a change in harmony or in key when listening to a piece of music.

Often regarded as an innate human ability, the automatic estimation of music content information, however, proves to be a highly complex task.

1.2 Scope of the Thesis

This thesis is concerned with the problem of extracting meaningful content information from music audio signals. Most of the previous works that address the problem of estimating musical attributes from the audio signal deal with these elements independently. However, when a musician analyzes a piece of music, his judgment is based on a global

musical context that encompasses various kinds of musical information. Musical elements are deeply related to each other and are analyzed in context. For instance, the chord progression is closely related to the metrical structure of a piece of music [Got01]: chords will change more often on strong beats than on other beat positions in the measure. It is also strongly related to the musical key: some chords are heard as more stable within an established tonal context [Kru90].

We believe that exploiting the interrelationship between musical attributes for their estimation should improve upon estimating them independently. This necessity has been underlined in the past. In [Tem99], Temperley and Sleator observe that:

"*[...] The idea, then, is to let the harmonic analysis influence the metrical analysis by favoring strong beats at changes of harmony. This presents a serious chicken-and-egg problem, however, since meter is crucial as input to harmony. One solution would be to compute everything at once, optimizing over both the metrical and harmonic rules, but we have not yet found an efficient way of doing this.*"

Our research concentrates on three musical descriptors related to the harmonic, the metrical and the tonal structure. More specifically, we focus on three musical attributes: the chord progression, the downbeats and the musical key. All of them are some of the most important attributes of Western tonal music.

The scope of this work is to develop a model that allows the joint estimation of the chords, the keys and the downbeats from polyphonic music recordings. We intend to show that integrating knowledge of mutual dependencies between several descriptors of musical content improves their estimation.

1.3 Relevant Music Theoretic Concepts and Terminology

Before going any further, we briefly review some musical concepts that are central to our thesis. This section aims at clarifying the music terminology that will be used in the following chapters. All musical concepts are understood here in the context of Modern Western music, *i.e.* after the 16^{th} century.

1.3.1 Notes

When an instrument produces a note, the human listener perceives a *pitch* that is a perceptual attribute of sound. In music, the term *note* is used to refer to the relative duration and pitch of a given sound. More details about the pitch will be given in Chapter 3.

1.3.2 Key and Scales

In western tonal music, pitches are governed by structural principles. The system of relationships between pitches corresponds to a *key*. A musical key implies a tonal center

that is the most stable pitch called the *tonic* and a *mode* (usually major or minor).

A musical scale is associated with each key. A *scale* is a series of notes arranged in ascending or descending order. Two consecutive notes are separated either by a tone (T), a semitone (S). The harmonic minor scale comprises also a $T + S$ interval. The position of tones and semitones within a scale associated to a key characterizes its mode.

Figure 1.1 represents a C major scale and its relative A *natural minor scale*. There are two common variations of the natural minor scale:

- the *harmonic minor scale*, in which the 7^{th} degree, both ascending and descending is raised a semitone ($G\#$ in Figure 1.1). We will consider this type of minor scale in Chapter 6.
- the *melodic minor scale*, in which the 6^{th} and the 7^{th} ascending degrees are raised a semitone($F\#$ and $G\#$ in Figure 1.1).

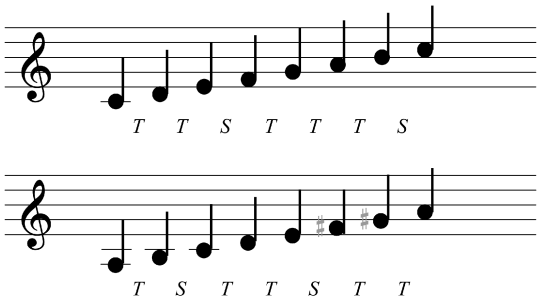

Figure 1.1: Example of major and minor scales: C major, A minor. The accidentals that characterize the harmonic and melodic minor scales are represented in grey.

In this work, we consider enharmonic equivalence, *i.e.* notes with different spelling but sounding the same are considered the same (C# is equivalent to Db). In Western tonal music, there are 12 pitches in an octave range. The major and minor scales and twelve tonic give rise to a total of 24 possible keys.

In a musical scale, the *tonic* or first scale degree (I) is the first note and it is the pitch upon which all other pitches of a piece are hierarchically referenced. The other scale degree, in the ascending order are: the *supertonic* (II), the *mediant* (III), the *subdominant* (IV), the *dominant* (V), the *leading tone* (VI) and the *subtonic* (VII). In the next chapters, we will refer in particular to the third and the fifth scale degrees, the *mediant* and the *dominant*, since the combination of these two notes plus the tonic corresponds to the triad formed on the tonic note, which is the most significant chord in a given key.

1.3.3 Chords

Chords that are specific to a key can be constructed around its scale. In Western tonal music, the chord progression determines the harmonic structure of a piece of music. It is strongly related to the musical key of the piece.

In this dissertation, a *chord* is defined as a combination of three or four notes sounded simultaneously. We include in this definition combinations of notes that sound nearly simultaneously, such as the arpeggio, which corresponds to an indivisible group of notes that are played one after the other. A succession of chords over time is called a *chord progression*.

Chords may be classified according to the number of notes they contain. Two-note combinations are called *dyads*, three-note combination are called *triads*. A chord is commonly characterized by its root note and by the intervals it contains. Classical triads are built from major and minor thirds, *i.e.* the distance between successive pairs of notes are 3 or 4 semi-tones. The major, minor, augmented and diminished chords are the most commonly used triads. Figure 1.2 illustrates the four basic triads based on the root-note C. Table 1.1 gives the relative semitone values for each triad.

Table 1.1: Compositions of the four basic triads computed on a root-note corresponding to a semitone value n.

chord	major	augmented	minor	diminished
root note	n	n	n	n
first third	(major) n+ 4	(major) n+ 4	(minor) n+ 3	(minor) n+ 3
second third	(minor) n+7	(major) n+ 8	(major) n+ 7	(minor) n+ 6

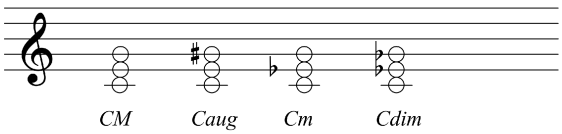

Figure 1.2: Example of common classical triads. From left to right: C major (C-E-G), C augmented (C-E-G#), C minor (C-Eb-G), C diminished (C-Eb-Gb).

Harmony is here understood as the system of structural principles governing the combination and the relationship between notes and chords.

In Western tonal music, the term *tonality* is often used to describe the relationships of melodies and harmonies relative to the tonic.

1.3.4 Metrical Structure

The metrical structure of a piece of music is a hierarchical structure. The meter is "the sense of strong and weak beats that arises from the interaction among hierarchical level of sequences having nested periodic components" [PEBB05].

- The most salient metrical level, called the *tactus* or *beat* level is a moderate level that corresponds to the foot-tapping rate.
- The *tatum* level corresponds to the "shortest durational values in music that are still more than accidentally encountered " [KEA06]. For instance, in Figure 1.3, the *tatum* level corresponds to the sixteenth notes.

- Musical signals are divided into units of equal time value called *measures* or *bars*.
- The relationship between measures and tactus/tatum is defined by the meter, which is usually indicated by a *time signature*, the number of units per measure.
- One important problem related to meter analysis is to find the position of the *downbeat* or the first beat of each measure.

The various metrical levels are illustrated in Figure 1.3.

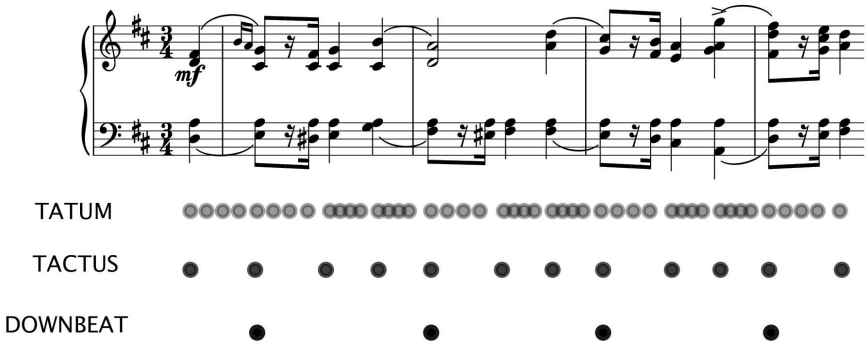

Figure 1.3: Illustration of the various metrical levels (extract of Schumann, *Kinderzenen*).

1.4 Applications

Within the context of Music Information Retrieval, many applications based on content-based indexing and retrieval have emerged, such as music classification, artist identification, mood classification or structural audio segmentation. These applications are mostly based on the use of musical descriptors that are extracted from the audio signal.

For instance, two different versions of the same underlying musical piece generally share a similar harmonic structure. The detection of cover versions is thus frequently based on the chord progression [SGHS08]. This can also be used for finding of *plagiarisms* [1]. For instance the main theme of French nursery rhyme *À vous dirais-je Maman* has been harmonized and used by several composers such as Mozart (piano Variations on *À vous dirais-je Maman* K. 265) or Liszt (*Années de Pèlerinage*).

The chord progression captures the characteristics of the accompaniment of musical pieces and their character. The knowledge of chord progression can thus be used for mood recognition, especially in popular music, for instance by measuring the ratio of major to minor chords in a piece of music. Information about the key can be used as well.

The automatic extraction of the harmonic structure may also be very useful to musicologists who can perform music analysis on large corpus of music pieces for which they

[1] A plagiarism is piece produced by a compositor by imitating another compositor's music while presenting it as one's original work.

may not have the score but only the recordings. It can also be used for the purposes of automatic composition.

Beat is fundamental to the perception of Western music. Beat/downbeat information can provide structural information about a live musical performance that may be used to make it interact with computer systems. Beat and downbeat tracking can be used for synchronizing a musical performance with some electronic devices such as electronic musical instruments or lights.

There are no limits to the range of possible applications of music content extraction. We thus believe that it is important to pursue efforts towards building rich models that can analyze music as musicians do.

1.5 Objectives

The objectives of this dissertation are listed bellow:

1. Review and analyze the previous approaches for chord progression, downbeat, global and local key estimation.
2. Compute audio features that capture the harmonic content of the signal and that will serve as an input to our model (without the need of an exact transcription).
3. Provide a reliable model for chord estimation that will serve as a baseline for studying the interrelationship with other musical attributes.
4. Provide a model that allows the joint estimation of the chords, the keys and the downbeats from polyphonic music recordings.
5. Consider complex cases of harmonic and metrical structure (variable meter, key changes).
6. Provide an analysis of our models through an evaluation over a large database of popular and classical music pieces.
7. Demonstrate that integrating knowledge of mutual dependencies between several descriptors of musical content improves their estimation.

1.6 Overview of the Thesis

This thesis is organized as follows. Figure 1.4 shows an overview of the interactions between musical attributes considered in the various chapters. In this dissertation, we consider harmony as a core around which other musical attributes are organized.

Chapter 2 - **Databases and Evaluation Measures Used in This Dissertation**. This chapter presents the evaluation methodology adopted along this thesis. In order to

avoid tedious repetitions of our evaluation methodology through the different chapters, we give in this chapter an overview of our evaluation test-sets and rules.

Chapter 3 - **Towards a Signal Representation for Harmonic Content Analysis**. This chapter investigates a number of typical representations of the audio signal in order to select the most appropriate one for the task of harmonic content analysis. We detail and explain the choice of the audio signal representation we use as an input to our model.

Chapter 4 - **Chord Progression Estimation From an Audio File**. This chapter concentrates on the problem of the automatic estimation of the chord progression from an audio file, using chroma features as observation of the music signal. From the audio signal, a set of chroma vectors representing the pitch content of the file over time is extracted. The chord progression is then estimated from these observations using hidden Markov models. Several methods are proposed that allow taking into account music theory, perception of key and presence of higher harmonics of pitch notes. They are evaluated and compared to existing algorithms through a large-scale evaluation on popular music songs.

Chapter 5 - **Joint Estimation of Chords and Downbeats**. This chapter presents a new technique for joint estimation of the chord progression and the downbeats from an audio file. A specific topology of hidden Markov models that enables modeling chord dependency on the metrical structure is proposed. This model allows us to consider pieces with complex metrical structures such as beat insertion, beat deletion or changes in the meter. The model is evaluated on a large set of popular music songs from the Beatles that present various metrical structures. We compare a semi-automatic model in which the beat positions are annotated, with a fully automatic model in which a beat tracker is used as a front-end of the system.

Chapter 6 - **Interaction Between Chords, Downbeats and Keys**. This chapter is concerned with the problem of key estimation. In a first part, we focus on the problem of global key estimation. Relying on previous works on key estimation, we extend the model presented in the previous chapter to a model for simultaneous downbeat, chord and key estimation from an audio signal. The model is evaluated on a set of popular music pieces. We then draw our attention to local key finding. We propose to address this problem by investigating the possible combination and extension of various approaches that have been previously proposed for global key estimation. The specificity of our approach is that we introduce key dependency on both the harmonic and the metrical structures. We evaluate and analyze our results on a new database composed of classical music pieces.

Chapter 7 -**Conclusion**. The last chapter of this dissertation summarizes the contributions of the present PhD work and proposes some perspectives.

1.7 Main Thesis Contributions

The principal contributions provided in this thesis are:

1. Chapter 3: An analysis and evaluation of several signal features extraction methods for harmonic content analysis of audio music.

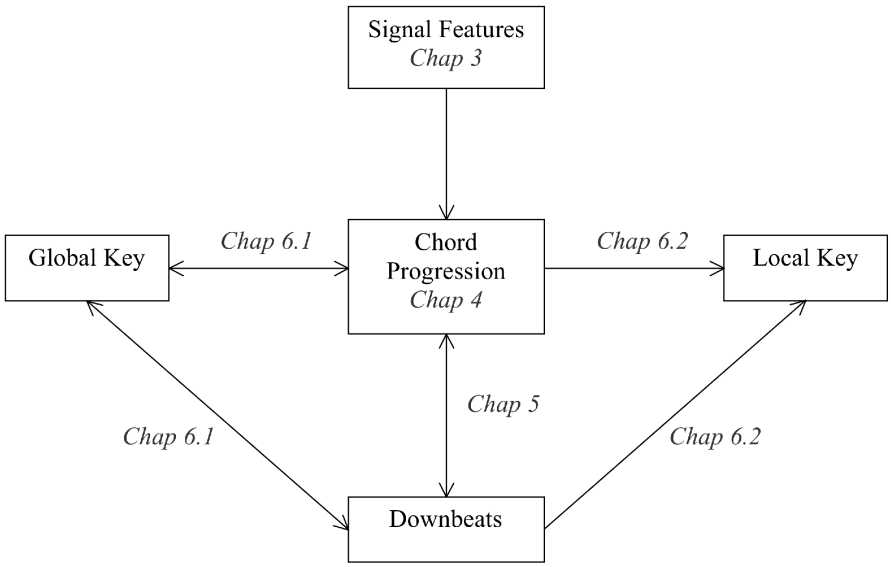

Figure 1.4: Interrelationships between chords, keys and downbeats considered in this dissertation.

2. Chapter 4: A model for the estimation of chords that encodes musical context information, takes into account the problem of harmonics in the signal, and does not need specific training.

3. Chapter 5: A model for the simultaneous estimation of chords and downbeats that exploit the interrelationship between these two musical attributes. We focus in particular on the problem of variable meter and imperfect beat tracking.

4. Chapter 6: A model for the simultaneous estimation of chords, main key and downbeats that exploits the interrelationship between these three musical attributes and an approach to local key estimation that is based on the harmonic and the metrical structure.

Chapter 2

Databases and Evaluation Measures Used in This Dissertation

This chapter presents the evaluation methodology adopted along this thesis. In the following chapters, we evaluate the performances of various models. For this, we rely on some evaluation measures and some test-sets that are common to all of the proposed systems. In order to avoid tedious repetitions of our evaluation methodology through the different chapters, we give in this chapter an overview of our evaluation test-sets and rules.

Contents

2.1 Introduction

Evaluation is an essential aspect in all areas of computational music analysis. This chapter is devoted to the evaluation methodology adopted along this thesis. In the following chapters, we evaluate the performances of various models. For this, we rely on some evaluation measures and some test-sets that are common to all of the proposed systems. In order to avoid tedious repetitions of our evaluation methodology through the different chapters, we give in this chapter an overview of our evaluation test-sets and rules.

Cross-references to this chapter will be made in the evaluation sections of the following chapters. We would advise the reader to start by having only a quick look at this chapter and come back when needed.

The chapter is divided into two main sections. Section 2.3 presents the various music collections used in this thesis to evaluate our work. In Section 2.4, we present and explain the evaluation measures used to measure the performances of our models on the test-sets.

2.2 About Evaluation

When designing a system that extracts some information from the audio signal, one must carefully evaluate the performances and the quality of the proposed models. In this dissertation, we are mainly concerned with two aspects of evaluation. On the one hand, we want to compare with each other several methods developed for a given task, or we want to measure whether certain changes in a given method lead to an improvement in the model performances. For instance, in Chapter 3, we compare several feature extraction methods and intend to select the best one among all. On the other hand, we are concerned with measuring the performances of a given method and measure its retrieval relevance. For instance, in Chapter 5, we aim to quantify the proportion of downbeat locations correctly estimated by our model.

2.3 Music Collections for Evaluation

In this section, we detail the characteristics of the databases used for evaluation of the models proposed in this thesis. In chapter 3, we conduct some experiments on two databases consisting of short excerpts of audio. In what follows, they are referred to as the *Signal Experiment test-set*. The rest of our work is mainly evaluated on two databases that are referred to as the *Beatles test-set* and the *Piano Mozart test-set*. These databases have been manually annotated in chords, keys, beats and downbeats either by previous researchers working on the same field, or by trained musicians, or by the author. This is described below.

2.3.1 Signal Experiment Test-set

The Signal Experiment test-set consists of a number of short excerpts of about 20 seconds extracted from audio recordings. It is divided into two subsets:

1. Non-percussive audio *DATClas* corresponds to audio excerpts of classical music with various instruments: string quartet, solo piano and orchestra.

2. Percussive audio *DATPop* corresponds to audio excerpts of popular and rock music that contains voices and drum sounds.

All the excerpts have been hand-labeled in chords by the author. The chords are annotated against a time grid defined by the beats. The detail of the excerpts is given in Table 2.1.

Table 2.1: Detail of the Signal Experiment test-set.

	Composer	Title
DATClas	Beethoven	String quartet Op. 127 1 ext1
	Beethoven	String quartet Op. 131 6 extract 1
	Beethoven	String quartet Op. 131 6 extract 2
	Mozart	Piano sonata KV 283 2 Andante CM extract 1
	Mozart	Piano sonata KV 309 1 CM extract 1
	Mozart	Piano sonata KV 310 1 Am extract 1
	Mozart	Piano sonata KV 310 1 Am Kempff extract 1
	Mozart	Piano sonata KV 310 1 Am Perahia extract 1
	Mozart	Piano sonata KV 310 1 Am Richter extract 1
	Beethoven	symphony no 5 extract 1
	Beethoven	symphony no 5 extract 2
DATPop	Beatles	Misery
	Beatles	Love Me Do extract 1
	Beatles	I Should Have Known Better extract 1
	Beatles	I m a Loser extract 1
	Beatles	Yesterday extract 1
	Beatles	Yesterday extract 2
	Enya	Caribbean blue extract 1
	Enya	Caribbean blue extract 2
	Queen	Lazing on a Sunday afternoon extract 1
	Queen	Lazing on a Sunday afternoon extract 2
	Shack	Natalies Party extract 1
	Shack	Natalies Party extract 2

2.3.2 Popular Music: The Beatles Test-set

In the MIR community, works related to chord estimation have almost exclusively been evaluated on the *Beatles test-set* since the chord labels annotations are freely available. This test-set is composed of 180 songs divided into 13 albums. All the recordings are polyphonic, multi-instrumental and contain drums and vocal parts. The list of the tracks and the corresponding albums can be found in Annex 7.2.

2.3.2.1 Chord Annotations

The chord annotations where kindly provided by C. Harte from the Queen Mary University of London[1]. This annotated test-set is by far the largest one available today.

The chords are annotated according to a special grammar proposed for chord labeling by Harte *et al.* in [HSAG05]. The annotation style that is adopted intends to be simple and intuitive to write and understand for musically trained individuals. The chords are defined by three parameters, the *root* note of the chord, the *quality* (component intervals that make up the chord relative to the root), and the *inversion* (degree of the chord played as its bass note). For instance, a C major chord will be annotated by $C : (3, 5)$, which reflects that it is a triad composed of a major third and a fifth, constructed on a root note of C. Shorthand labels for common chords are also proposed.

The original chord annotations have been obtained either from listening to the audio or from music scores and they correspond to the exact transcription of the chords that are played. They thus present a large variety of chord labels including some complex chords such as major and minor 6^{th}, 7^{th} or 9^{th}.

We aim to compare the output of our algorithm with the ground-truth annotations. Since our chord lexicon is composed only of major and minor triads, we have performed a mapping from complex chords in the annotation to their root triads. This point is important when analyzing the results. For instance, a Dm7 (D-F-A-C) chord is considered as a Dm chord (D-F-A). The augmented chords, which include a major third, have been mapped to major chords whereas the diminished chords, which include a minor third, were mapped to minor chords.

Analysis of the complete set of the Beatles test-set has shown that most of the chords correspond to major and minor triads. It was found in [MDH+07] that major chords prevail, accounting for 76% of all chords, whereas the minor chords account for 24%.

2.3.2.2 Key Annotations

We select 55 Beatles songs from the first eight albums for which we assigned a global key. We select songs that remain in the same key from the beginning to the end so that there are no modulations. The list of the songs with the corresponding global keys is given in Table 2.2. This subset of the complete *Beatles test-set* will be referred to as the *Beatles test-set_key* in the following.

2.3.2.3 Metric Structure Annotation

The tactus, tatum and downbeat positions of the Beatles songs were manually annotated by the author and checked by trained musicians.

It has been annotated using the Open Source tool *Wavesurfer*[2] placing on-the-fly

[1] www.elec.qmul.ac.uk/digitalmusic/
[2] www.speech.kth.se/wavesurfer/

Table 2.2: Beatles songs annotated in global key: *Beatles test-set_key*.

Album	Title	Key
Please Please Me	01 - I Saw Her Standing There	EM
	03 - Anna (Go To Him)	DM
	04 - Chains	A#M
	05 - Boys	EM
	06 - Ask Me Why	EM
	07 - Please Please Me	EM
	08 - Love Me Do	GM
	09 - P. S. I Love You	DM
	13 - There is A Place	EM
With The Beatles	01 - It Won t Be Long	EM
	02 - All I' ve Got To Do	EM
	03 - All My Loving	EM
	05 - Little Child	EM
	06 - Till There Was You	FM
	07 - Please Mister Postman	AM
	08 - Roll Over Beethoven	DM
	09 - Hold Me Tight	FM
	12 - Devil In Her Heart	GM
	13 - Not A Second Time	GM
A Hard Days Night	02 - I Should Have Known Better	GM
	03 - If I Fell	DM
	05 - And I Love Her	EM
	06 - Tell Me Why	DM
	08 - Any Time At All	DM
	11 - When I Get Home	AM
	12 - You Can t Do That	GM
Beatles For Sale	01 - No Reply	CM
	02 - I am a Loser	GM
	04 - Rock and Roll Music	AM
	05 - I will Follow the Sun	CM
	06 - Mr. Moonlight	F#M
	07 - Kansas City- Hey, Hey, Hey, Hey	GM
	08 - Eight Days a Week	DM
	09 - Words of Love	AM
	11 - Every Little Thing	AM
	13 - What You are Doing	DM
Help	02 - The Night Before	DM
	04 - I Need You	AM
	08 - Act Naturally	GM
	09 - It s Only Love	CM
	10 - You Like Me Too Much	GM
	12 – I' ve Just Seen a Face	AM
Rubber Soul	01 - Drive My Car	DM
	05 - Think For Yourself	GM
	11 - In My Life	AM
	14 - Run For Your Life	DM
Revolver	08 - Good Day Sunshine	BM
	09 - And Your Bird Can Sing	EM
	10 - For No One	BM
	13 - Got To Get You Into My Life	GM
Sgt Peppers Lonely Hearts Club Band	02 - With A Little Help From My Friends	EM
	04 - Getting Better	CM
	05 - Fixing A Hole	FM
	09 - When I m Sixty-Four	DbM
	12 - Sgt. Pepper s Lonely Hearts Club Band (Reprise)	DM

markers while listening to the music. Markers have then been manually corrected in order to correct the inherent software latency.

Meter information for each song was provided by the American musicologist Alan W.

Pollack[3]. The original set comprises 180 songs of the Beatles. We reduced it to 165 songs removing songs having an overcomplicated metric structure and containing parts where downbeats were perceptually ambiguous and were extremely difficult to predict and annotate, even for a trained musician. For instance, the song *Good Morning, Good Morning* was not analyzed because, according to A.W. Pollack, the meter is "4/4 in intro, bridge and outro; anything but predictable in verse". For this reason, those files were not annotated.

The songs of the test-set can be classified according to their metric structure in the following way:

- 8 songs are in 3/4 meter
- 9 songs have a variable meter (presenting at least one change in time signature, more than two for most of them)
- 25 songs present some insertion or deletion of beats (insertion of a measure with unexpected time-signature in a constant meter passage that does not musically correspond to a change in the meter.)
- The rest of the songs have a constant 4/4 meter.

The detail of those songs is given in Table 7.4.

Table 2.3: Evaluated songs that have a particular metric structure.

Meter	Title (album/song number)
3/4	Baby's In Black (4/3) You've Got To Hide Your Love Away (5/3) Norwegian Wood (This Bird Has Flown) (6/2) She's Leaving Home (8/6) Long, Long, Long (11/7) Oh! Darling (1/4) Dig A Pony (13/2) Dig It (13/5)
variable	A Taste Of Honey (1/12) Lucy In The Sky With Diamonds (8/3) Being For The Benefit Of Mr. Kite (8/7) Strawberry Fields Forever (9/8) All You Need Is Love (9/11) Happiness Is A Warm Gun (10/8) I Want You (She's So Heavy) (12/6) Two Of Us (13/1) I Me Mine (13/4)

2.3.3 Classical Music: The Piano Mozart test-set

The *Piano Mozart test-set* was introduced for the purpose of evaluating the performances of the local key algorithm. We are not aware of any available test-set that contains

[3]http://www.icce.rug.nl/∼soundscapes/DATABASES/AWP/awp-notes_on.html

pieces annotated in local keys. We decided to annotate some Mozart piano pieces for two reasons. First they are interesting from the point of view of local key because they contain many modulations. Secondly, it was easier to annotate these pieces than others because the author is very familiar with them. The test-set consists of 5 movements of Mozart piano sonatas listed in Table 2.4 corresponding to about 30 minutes of audio music.

Table 2.4: The *Piano Mozart test-set*.

Reference of the piano sonata	movement
KV 283	1
KV 283	2
KV 309	1
KV 310	1
KV 311	2

The author and two other trained musicians from the Musikhorshule of Karlsruhe (Germany) manually annotated the chord and key progression ground truth. First, a list of the chords and keys with their duration in beats has been provided. Beat and downbeat locations were annotated by hand with the help of the software *Wavesurfer*. Then, the list was automatically mapped to the annotated beat locations resulting in the ground truth we use. The pieces have been annotated in mostly by ear but also relying on the scores when ambiguities were found.

It has to be noticed that it is very hard to label Mozart pieces in chords and musical keys, even for a well-trained musician because on the one hand, there are a lot of ornamental notes (such as appoggiaturas, suspensions, passing notes etc.) and on the other hand, harmony is frequently incomplete (some notes of the chord are missing). This makes the choice of chord labels very difficult. Changes from one key to another are often ambiguous, in particular when they are very short. Moreover, modulation is very often a smooth process, it can take several bars to establish properly a tonal center. Segments corresponding to transition from one key to another have been labeled as transition parts and are ignored in the evaluation.

2.3.4 Databases Used for Evaluation in Each Chapter

- In Chapter 4, we compare and evaluate several chord estimation algorithms using the first eight albums of the *Beatles test-set*. This corresponds to a total of 110 songs.

- In Chapter 5, we evaluate our chord/downbeat simultaneous estimation model using a subset of 165 of the 180 songs of the *Beatles test-set*. The songs that have not been used are referenced in Table 2.5.

- In Chapter 6, we evaluate the model for simultaneous chords, downbeats and global key on the 55 Beatles songs annotated in global key and described above. We evaluate our local key estimation algorithm on the *Piano Mozart test-set*.

Table 2.5: Beatles songs not considered in the evaluation in Chapter 5.

Album	Song number
7	87 - 91 - 97
8	105 - 107 - 108
10	124 - 127
11	140 - 141 - 142 - 150
12	158
13	171 - 180

2.4 Evaluation measures

In this section, we present the measures that have been used for evaluating the various algorithms that we have implemented. The chord, key and downbeat results discussed and analyzed in the next chapters (Chapters 4, 5 and 6) have been obtained relying on these measures. For each musical attribute considered, we evaluate the performances of our model by comparing the estimation (output of our algorithm) with the ground truth (human manual annotations).

2.4.1 Beat and Downbeat Tracking Evaluation Measure

In Chapter 5, we evaluate the performances of our downbeat tracking model. We also evaluate the performance of a beat tracker that is used as a front-end of our system. For this, we compare the beat/downbeat times of our system output with the hand-labeled beat/downbeat times that are considered as the correct beat/downbeat locations (see Figure 2.1).

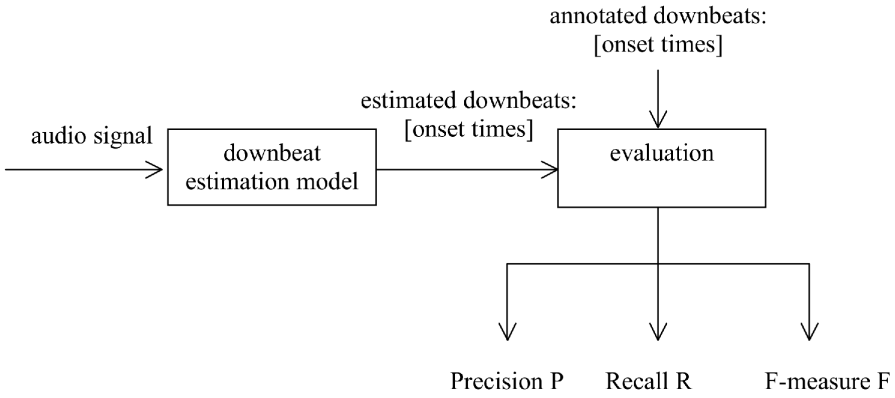

Figure 2.1: Overview of the beat/downbeat evaluation measure.

It is important to notice that it is very difficult to annotate beat and downbeat locations in an objective manner since it is a perceptual concept. Human experts may in particular disagree on the downbeat locations when the structure of the music piece is complex. As underlined above, this is one of the main reason why we do not use the entire *Beatles test-set* for downbeat tracking evaluation.

A large number of evaluation measures for beat/downbeat tracking have been pro-

posed in previous works. We refer the reader to [Dav07] for a detailed review. In this work (chapter 5), the evaluation is performed using the standard Precision, Recall and F-measure. This measure has been previously used by Dixon in [Dix06].

- *Precision* P is defined as the ratio of relevant retrieved beat/downbeat positions from the total of retrieved ones.
- *Recall* R is defined as the ratio of relevant retrieved beat/downbeat positions from the total of relevant positions.
- The *F-measure* F combines the two using the ratio of their geometric to arithmetic mean: $F = \frac{2RP}{R+P}$.

An estimated beat position is considered as correct if it is within a given tolerance window of the ground truth time.

Following [Pee09], the tolerance window w is defined as 10% of the minimum (over time) distance between two successive beats in the track. It is centered on the estimated beats when computing the Precision and centered on the annotated beats when computing the Recall. The tolerance window depends on the local tempo (distance between two beat markers) in order to avoid drawing misleading conclusions from the results. Indeed, a fixed tolerance window of 0.166 s for instance would be very restrictive for slow tempi (half-beat duration of 0.5 s at 60 bpm) but would mean accepting counter-beats as correct for fast tempi (half-beat duration of 0.166 s at 180 bpm).

Let c denote the number of correct beat/downbeat detections, f^+ the number of false positive (unmatched reported beat times, *i.e.* beats estimated outside of any of the tolerance windows) and f^- the number of false negative (unmatched correct beat times, *i.e.* misses), the Precision, Recall and F-measure can be expressed as following:

$$P = \frac{c}{c + f^+}$$
$$R = \frac{c}{c + f^-}$$

The beat/downbeat evaluation measure is illustrated in Figure 2.2.

2.4.2 Chord Evaluation Measures

We aim at comparing the output of our chord estimation algorithm with the ground-truth annotations. As stated above, since our chord lexicon is composed only of major and minor triads, we have performed a mapping from complex chords in the annotation to their root triads. We consider two aspects of chord estimation: the label accuracy *i.e.* how the estimated chord is consistent with the ground truth (Chapters 4, 5 and 6) and the segmentation accuracy i.e. how the detected chord changes are consistent with the actual locations (see Chapter 5). In Figure 2.3, we provide an overview of the chord evaluation measure.

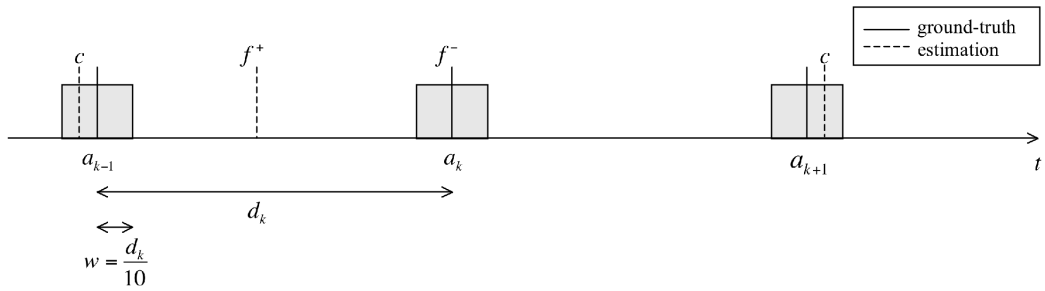

Figure 2.2: Illustration of beat/downbeat evaluation measure. The ground-truth positions are indicated by solid lines and the estimated beat positions by dashed lines. a_{k-1}, a_k, a_{k+1}: annotated beats, c: correct estimation, f^+: false positive, f^-: false negative, d_k: duration between two annotated beats.

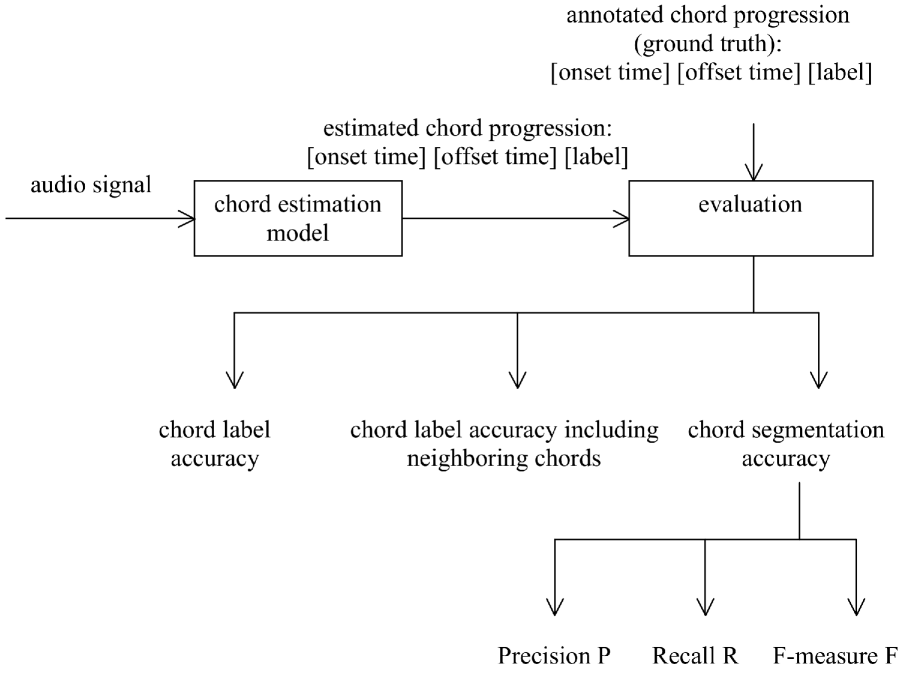

Figure 2.3: Overview of the chord evaluation measure.

2.4.2.1 Label Accuracy

The *chord label accuracy* measure is illustrated in Figure 2.4 and is defined as follows.

For each song s of the test-set, let $T_A = (t_{A1}, t_{A2}, \ldots, t_{AM})$ denote time positions corresponding to the annotated (ground truth) chord changes and let $T_E = (t_{E1}, t_{E2}, \ldots, t_{EN})$ denote time positions corresponding to the estimated chord changes. We note $T = T_A \cup T_E$. We note $\{T_k = [t_k, t_{k+1}]\}$ the series of segments defined by this union. Each segment $[t_k, t_{k+1}] \subset T$ has a length d_k. We note $\hat{C}_k$ (C_k) the estimated (annotated) chord over T_k. The chord estimation rate μ_s is computed as:

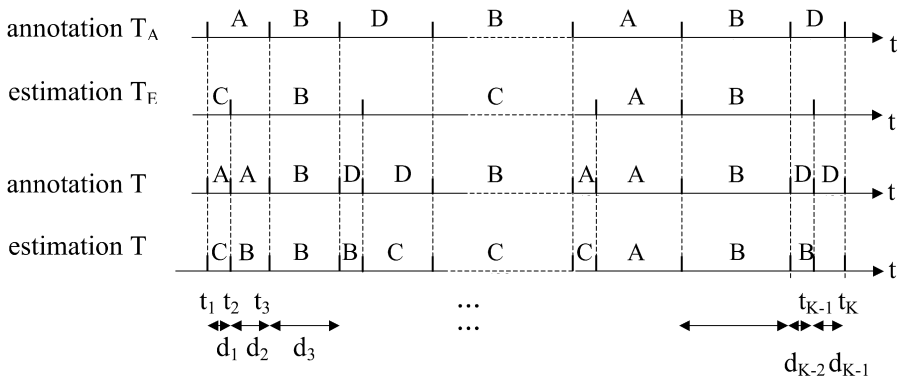

Figure 2.4: Illustration of the chord label accuracy measure.

$$\mu_s = 100 * \frac{\sum_{k \text{ so that } C_k = \hat{C}_k}^{K-1} d_k}{\sum_{k=1}^{K-1} d_k} \tag{2.1}$$

In this dissertation, the chord estimation rate μ_s will be referred to as the *chord label accuracy*. Note that in this study, we do not consider "non-existing chords", noted "N" in the annotation (denoting noise, silent parts or non-harmonic sounds). They are counted as errors in the evaluation.

The chord estimation results we give in the next chapters correspond to the average of the values corresponding to the mean and standard deviation of correctly identified frames per song, computed across all the songs belonging to the test-set.

2.4.2.2 Segmentation Accuracy

The *chord segmentation accuracy* is evaluated using a measure similar to the one chosen for downbeat evaluation. We use the standard Precision P (ratio of detected chord changes that are relevant), Recall R (ratio of relevant chord changes detected) and F-measure F, using a tolerance window w of 30% of the minimum distance between two beats in the track. w is chosen to be larger than the one used for downbeat evaluation but below the tatum period.

2.4.2.3 Neighboring Chords Confusions

We will also refer to the chord estimation results considering neighboring triad confusions. Harmonically close chords are in general neighbors on the circle of fifths (see Chapter 6). The hierarchy between chords presents some similarities with the relationships within keys. We thus follow for chord estimation the procedure adopted during the MIREX 2005 Key estimation contest, where keys were considered as close if they had one of the following

relationships: distance of fifth, relative minor and major and parallel *i.e.* having the same tonic but different mode (major or minor). Chords are here considered as harmonically close if they have one of the particular relationships described in Table 2.6.

Table 2.6: Example of particular relationships between a C major chord and other chords. Weights attributed to neighboring chords in comparison with MIREX 2005 key estimation task.

Reference chord	C major	weight chord	weight key (MIREX 2005)
Relative	Am	1	0.3
Parallel	Cm	1	0.2
Dominant	GM	1	0.5
Subdominant	FM	1	0.5

2.4.3 Keys

2.4.3.1 Main Key

In the first part of Chapter 6, the key estimation evaluation is performed using an 8-fold cross-validation. The test-set is divided into eight parts according to the albums and each part is evaluated using the seven remaining parts as training data. We indicate the rate of correct estimation using two evaluation measures:

- EE (exact estimation) indicates the percentage of exactly estimated key,
- ME (MIREX estimation) gives the estimation rate according to the measure proposed for the MIREX 2005 key estimation task. Neighboring keys are taken into account (see Table 2.6) and the score is obtained using the following weights: 1 for correct key estimation, 0.5 for perfect fifth relationship between estimated and ground-truth key, 0.3 if detection of relative major/minor key, 0.2 if detection of parallel major/minor key.

For an overview of the main key evaluation measure, see Figure 2.5.

2.4.3.2 Local Keys

In the second part of Chapter 6 devoted to local key estimation, we consider, as in [CV05], two aspects of the results: the *key label accuracy i.e.* how the estimated key is consistent with the ground truth, and the *key segmentation accuracy i.e.* how the detected modulation points are consistent with the actual locations. The *local key label accuracy* evaluation measure is the same as the *chord label accuracy* evaluation measure used for chord estimation.

The *key segmentation accuracy* is expressed with the Precision, Recall and F-measure. Key changes are not abrupt and often last several bars. Two established keys are often separated by a transition part where no key is firmly established. These parts, which have been labeled as transition parts T in the ground truth, need to be taken into account in

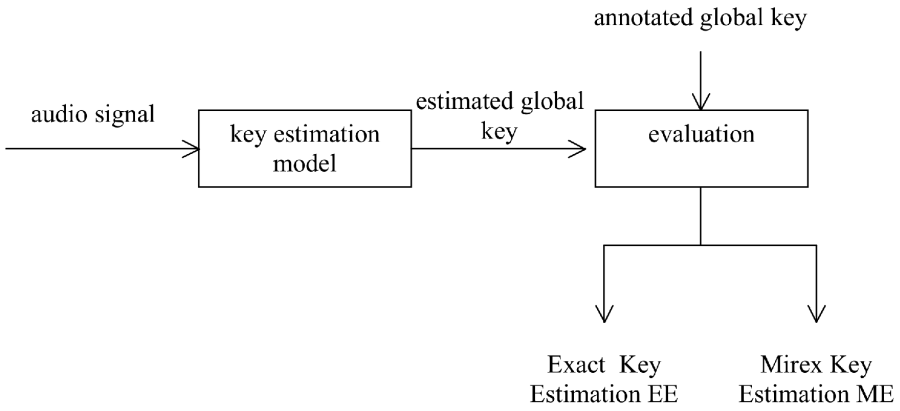

Figure 2.5: Overview of the main key evaluation measure.

the evaluation of segmentation accuracy. The tolerance window chosen in the case of local key estimation is thus larger than in the case of chord estimation: we present in Chapter 6 results with w corresponding to 1 or 2 bars.

2.4.4 About Statistical Significance Testing

During our experiments, we will use *paired samples t-test* (or *dependent samples t-test*) in order to compare the various methods we propose. They will be used to measure whether if the changes in the results from one method to another are statistically significant or not. Paired samples t-test is a statistical technique that allows the comparison between two population means when the two samples that are correlated.

2.4.5 About Evaluation of Algorithms Based on Training

In this dissertation, we evaluate some algorithms based on training: in Chapter 4, Sections 4.4.3.1 and 4.4.4.4, the chord model parameters are trained on a labeled database, as well as the key-dependent chord transition matrix proposed in Chapter 6, Section 6.3. These algorithms are evaluated on the *Beatles test-set*. Let k denote the number of albums considered in the test-set. The algorithms are evaluated using a k-fold cross-validation. The test-set is divided into k parts according to the albums and each part is evaluated using the $k-1$ remaining parts as training data.

This procedure is adopted in order to avoid the so called "album effect" [WFS01] [KWP06]. A given album is generally recorded within a short time period and songs from the same album are likely to share common spectral characteristics (choice of instrumentation, audio post-production, etc.), whereas variation in the artist's musical style over the year may vary more between albums. This is why we use complete albums as training while the others are used as testing.

Chapter 3

Towards a Signal Representation for Harmonic Content Analysis

This chapter investigates a number of typical representations of the audio signal in order to select the most appropriate one for the task of harmonic content analysis. We explore several schemes for chromagram computation and investigate several issues related to the use of each representation. We detail and explain the choice of the audio signal representation we use as an input to our model.

Contents

3.1 Introduction

In this dissertation, we are interested in estimating various musical attributes that are centered on the chord progression. We work directly on the audio signal. In computational music analysis, the first step of any algorithm that works on audio is to extract a set of feature vectors that represent the signal.

Harmonic analysis of a piece of music is a problem that has interested musicologists for centuries. Harmonic analysis from a music score is a complex problem, but it is even more complicated when working directly on the audio signal. Indeed, in addition to the difficulties inherent to the musical syntax (grammar, language), the first difficulty is to obtain information about the pitches of the notes that are present in the audio signal.

The aim of this chapter is to investigate a number of possible representations of the audio signal in order to select the most appropriate one for the task of harmonic content analysis. Our goal is to provide signal features that are suitable for the chord estimation task. We do not attempt to propose a new signal feature extraction technique but we justify the choice of the input representation we use in our system. Many *chroma*-based signal representations that capture the harmonic content of an audio signal have been proposed in the past. However, little time has been devoted to the comparison and the evaluation of these approaches. In this chapter, we concentrate on this point. The major contributions of this chapter are the following:

1. We review several methods for extracting *chroma* features from the audio signal.

2. We focus on the problem of evaluating and comparing the various representations and propose new evaluation methods.

3. We annotated in chords a database consisting of a number of short excerpts of classical and popular music.

4. We compare the various representations on this database providing statistical tests to enhance our analysis.

5. We investigate the use of beat-synchronous *chroma* features for harmonic content analysis.

Organization of the chapter:

This chapter is organized as follows. In Section 3.2, we review some basic concepts of audio signal processing. We then introduce in Section 3.3 the notion of *chroma* and propose several methods for chromagram computation in Section 3.4. We analyze two problems related to the use of chroma features for harmonic content analysis in Section 3.5. The various methods are evaluated in Section 3.6. A conclusion closes this chapter.

3.2 A Representation of Audio for Harmonic Content Analysis

3.2.1 Music Transcription-Based Approaches

Reproducing the human capability of analyzing tonal and harmonic structure of a piece of music with computers is an ambitious challenge. The most straightforward way to recreate the human process of music analysis is to start automatic analysis from a symbolic representation. In the scope of harmonic and tonal analysis, some efforts have been initially devoted to the analysis of chord and key sequences using MIDI representation of music [Tem05] [TS99]. In particular, some tools that allow tonal and harmonic analysis of music in the symbolic domain, have been developed.

The Melisma Music Analyzer, developed by D. Temperley & D. Sleator is a system for analyzing music and extracting information from it. The analyzer takes a piece represented as an "event list" that is a list of notes, with pitch, on-time, and off-time (MIDI files can be used as input as well). It extracts information about meter, phrase structure, contrapuntal structure (the grouping of notes into melodic lines), harmony, pitch spelling and key. The HARMONY program produces a harmonic analysis consisting of a series of segments labeled with roots and a spelling assigned to each pitch-event. Finally, the KEY program produces a key analysis, consisting of a series of sections labeled with keys and (optionally) a Roman numeral analysis showing the function of each chord relative to the current key. The main goal in this project has been to develop models of musical cognition. The components of the Melisma system are all based on the concept of preference rules.

OpenMusic is a visual programming language based on CommonLisp / CLOS developed at IRCAM. It provides classes and libraries that make it a very convenient environment for music composition and analysis. Different representations of a musical process are handled, among which common notation, midi piano-roll and sound signal. A symbolic representation of a chord progression can be analyzed with OpenMusic, but it requires information about the key signature and about chord segmentation. Chords are treated as the combination of discrete tones and recognized from the result of polyphonic analysis based on music theory.

3.2.2 Chroma Representation, an Alternative to Transcription

The work conducted in the symbolic domain could be applied to audio signals using a symbolic transcription. However, the symbolic transcription (the score) of a piece of music is not always available, especially in music where there is a large part devoted to improvisation such as jazz music. In addition to that, algorithms that extract a transcription from an audio signal are still limited and costly.

A number of recent works have shown that it is possible to accurately extract a music description of the signal without relying on a symbolic representation. An intermediate between low-level signal features and symbolic representation can be used to extract some

musical attributes such as the chord progression. Since their introduction in 1999, *Pitch Class Profile* (PCP) [Fuj99] or *chroma*-based representations [Wak99] have become a common feature for estimating chords and musical keys from audio recordings, as well as for conducting audio similarity retrieval tasks.

3.2.2.1 Definition

Shepard reported in the 1960s that our perception of pitch is two-dimensional and can be modeled by a helix (see Figure 3.1). He noticed that the representation of pitch into a helical curve is quite ancient since it had previously been proposed by Drobisch in 1846. This helix is characterized by two attributes: i) the *Tone Height* or over-all pitch level (octave number), that corresponds to the vertical axis, and ii) the *Chroma* that corresponds to the angle. By dividing the base of the helix into 12 equal parts, we can obtain the 12 pitches of the equal-tempered chromatic scale.

Two notes a number of octaves apart (for instance the C1 and C2 notes) will share the same rotation on the chroma circle represented at the base of the helix shown in Fig 3.1. In music theory, the term *pitch class* is rather used than chroma.

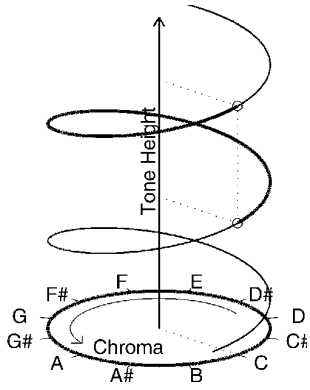

Figure 3.1: Shepard's helix of pitch perception, adapted from [BW05].

Chroma/Pitch Class Profile features are traditionally 12-dimensional vectors, with each dimension corresponding to the intensity associated with one of the 12 semitone pitch classes (chroma) of the Western tonal music scale, regardless of octave.

In the rest of this dissertation, we will assume that the order of the pitch classes in a PCP/chroma vector is: $C, C\#, D, \ldots, A\#, B$. We will often refer to each pitch class using a number: 1 corresponds to C, 2 corresponds to $C\#$, and so on until 12 that corresponds to B.

The temporal sequence of chroma vectors over time is known as *chromagram*. Conceptually, the chromagram is a frequency spectrum folded into a single octave. Chroma features are closely related to the music signal and working with them is very convenient when dealing with problems related to harmony or tonality. Pooling the spectrum into twelve bins that correspond to the twelve pitch classes of the equal-tempered scale results in a signal representation that allows identifying pitches by an octave. As emphasized

in [EP07], the chroma features capture both melodic information (since the melody note will typically dominate the feature) and harmonic accompaniment information (since other notes in chords will result in secondary peaks in a given chroma/PCP vector). The use of such a mid-level representation overcomes the problem of automatic transcription.

Before going into details in the chromagram representation, we review and compare some classical time-frequency representations of the signal. They will serve as a basis for chromagram computation.

3.2.3 Representation of Music Signals, Notations

We rely on the common assumption that the music signal is stationary (i.e., its statistical properties do not vary with time) in a very short time duration and thus that we can consider music sounds as nearly periodic signals. This means that the waveform repeats itself, in a slightly modified version, at a regular time interval that is called the *period*. The reciprocal of the period of the signal is called the *fundamental frequency* and denoted by $f0$.

When an instrument produces a sound, the human listener perceives a pitch that is a perceptual attribute of the sound related to the fundamental frequency. The pitch is a subjective quality of the sound often described as highness or lowness. Pitched instruments also include certain percussion instruments, such as the marimba, the vibraphone, the tubular bells or the timpani. Non-harmonic sounds for which the pitch is undefined, such as the cymbals, the gongs, or the tam-tams make sounds rich in inharmonic partials [1]. These sounds do not belong to the harmony progression of a piece of music. Musicians associate music notes symbols to the pitches.

According to Fourier's theory, a periodic signal can be approximated by a finite sum of sinusoids whose frequencies are integer multiple of the fundamental frequency and whose magnitude and phase can be uniquely determined to match the signal. The frequency of each sinusoid is called *harmonic*. In general, the harmonics of music sounds do not have frequencies that are exactly multiples of its $f0$. For this reason, they are often called *partials*. In this thesis, we are interested in sounds of which the partials are nearly harmonically related. They are called harmonic sounds.

Let us define some notations that will be used in the rest of the chapter. A music signal $s(t)$ will be understood as a superimposition of N_n individual notes $n_i, i \in [1 : N_n]$ produced by musical instruments. Each note is characterized by its perceived pitch of frequency f_0 and a finite and small number K of partials of frequencies $f_k = kf_0, k \in [1 : K]$, of amplitude a_k. The spectral pattern composed of the series of partials characterizes the sound perceived by the human listener. A quasi-periodic music signal $n(t)$ corresponding

[1] The inharmonic partials correspond to partials that deviate from their expected position according to the harmonic model described above. They can also be observed in the string instrument sounds such as the piano.

to a single note can thus be expressed as:

$$n(t) = \sum_{k=0}^{K} a_k(t) cos(2\pi f_k t + \Phi_k) \; with \; f_k = k f_0 \tag{3.1}$$

where a_k and Φ_k correspond to the amplitude and phase of the various sinusoids that approximate the signal.

A harmonic signal $\tilde{s}(t)$ that is a superimposition of N_n individual notes can then be expressed as:

$$\tilde{s}(t) = \sum_{n=1}^{N_n} \sum_{k=0}^{K} a_{k,n}(t) cos(2\pi k f_{0,n} t + \Phi_{k,n}) \tag{3.2}$$

Equations (3.1) and (3.2) give the expression of an ideal music signal composed of a set of exactly harmonically related sinusoids. In practice, the observed signal $s(t)$ contains some components that are not explained by the sinusoids, for instance the background noise or the inharmonic partials. It can be expressed as a sum of harmonic components $\tilde{s}(t)$ plus a residual $\epsilon(t)$ that comes from the unexplained components:

$$s(t) = \tilde{s}(t) + \epsilon(t) \tag{3.3}$$

$$= \sum_{n=1}^{N_n} \sum_{k=0}^{K} a_{k,n}(t) cos(2\pi k f_{0,n} t + \Phi_{k,n}) + \epsilon(t) \tag{3.4}$$

The ratios between the first partials of a music sound and the fundamental frequency approximately correspond to musical intervals. In Table 3.1, we represent the musical intervals corresponding to the ratios between the 6 first partials and the fundamental frequency of a C note.

Table 3.1: Intervals between the first 6 partials of a complex tone and its fundamental frequency $f0$. Example for the partials of a C note.

Pitch class	Partial	Frequency	Approximate interval with f_0
C	1	f_0	unison
C	2	$2 * f_0$	octave
G	3	$3 * f_0$	octave + 5^{th}
C	4	$4 * f_0$	2 octaves
E	5	$5 * f_0$	2 octave + major 3^{rd}
G	6	$6 * f_0$	2 octave + 5^{th}

3.2.4 About Acoustic Signal Representation

Algorithms for the automatic analysis of audio music signals rely in general on a spectral representation of the signal. The discrete Short Time Fourier Transform (STFT) is the most commonly used representation. Although it is very popular, a shortcoming of

this representation is that the frequency components are equally spaced and thus have a constant resolution, which implies that a global compromise between time and frequency resolution has to be made. Multi-resolution approaches have been proposed as an alternative to the Fourier transform.

3.2.4.1 Fourier Transform

Since its introduction in the 18th century, the Fourier transform and its extensions have become the most common signal representation used in signal processing.

In signal processing, we process finite extent signals. The Discrete Fourier Transform (DFT) computes a discrete-frequency spectrum from a discrete-time signal of finite length:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N} \tag{3.5}$$

where $x(n)$ denotes the input signal at time sample n, $k = 0, 1, \ldots, N-1$ denotes the frequency bin index and $X(k)$ denotes the k^{th} spectral sample.

The computation cost of a DFT can be very expensive. A much faster algorithm has been developed by Cooley and Tukey in 1965 [CT65], called the Fast Fourier Transform (FFT) and is used in general in signal processing.

In practical signal processing, a window $w(n)$, that is, a weighting function, is applied to data to reduce the undesirable effects related to spectral leakage associated with finite observation intervals [Har78]. The Short Time Fourier Transform (STFT) represents the frequency content of a short segment (of limited duration) of the signal. This segment of limited duration is assumed to be stationary. The STFT of a discrete signal $x(n)$ can be calculated as:

$$X(k) = \sum_{n=0}^{N-1} w(n) x(n) e^{-j2\pi kn/N} \tag{3.6}$$

where $w(n)$ is the temporal window function and $k = 0, 1, \ldots, N$ denotes the frequency bin index.

The length of the window N determines the time and the frequency resolution. The accuracy in the frequency domain will increase with the length of the window. However, this occurs at the expense of the time resolution. Moreover, as the window length increases, the assumption of the stationarity of the signal during the analysis segment becomes weaker.

3.2.4.2 Frequency Resolution Versus Time Resolution

When analyzing music signals, the choice of the length of the analysis window is a key consideration. It determines the trade-off of time versus frequency resolution which affects the smoothness of the spectrum and the detectability of the sinusoidal components. On the one hand, good temporal resolution and therefore a short window length are needed

in order to detect fast changes in the signal (such as note onsets for instance). On the other hand, a large analysis window is necessary to provide the required frequency resolution so that closely spaced sinusoids, corresponding to adjacent pitches, can be resolved.

Let us consider an audio music signal with two sinusoids of frequencies f_1 and f_2 that correspond to adjacent pitches. We note $\Delta f = f_2 - f_1$. In a music signal, when two sinusoids corresponding to adjacent pitches have nearby frequencies separated by Δf Hz, it is necessary that the window length N is large enough so that the spectrum exhibits two peaks (see Figure 3.2).

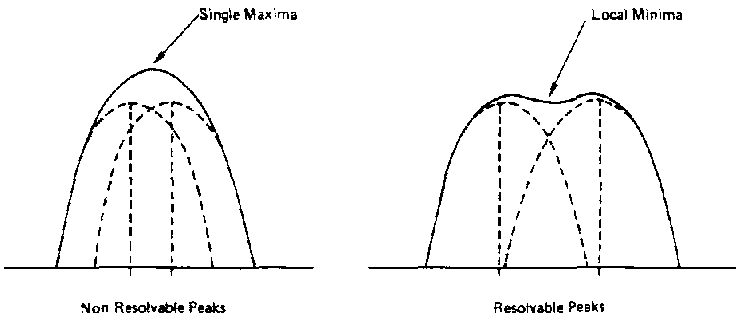

Figure 3.2: Spectral resolution of nearby peaks. From [Har78]. Left: non-resolvable peaks. Right: resolvable peaks.

According to [Smi08], the lower bound for the minimum FFT length N is:

$$N \geq K_w \frac{f_s}{\Delta f} \tag{3.7}$$

where K_w is a constant that depends on the window function main-lobe width. Table 3.2 gives the main-lobe width in-bins, K_w, for various windows. The minimum resolving window length can be determined using the sharper bound K_w^* empirically found [AS04].

Table 3.2: Main-lobe width in-bins K_w and minimum effective values K_w^* for various windows. From [Smi08].

Window Type	K_w	K_w^*
Rectangular	2	1.44
Hamming	4	2.22
Hann	4	2.36
Blackman	6	2.02

Because of the logarithmic scaling of the Western tonal music scale, pitch frequencies are closer in lower frequencies. Two adjacent notes tuned in equal temperament form a semitone and are separated by 6% of the frequency of the lowest note. Indeed, let f_k and f_{k+1} denote respectively the frequency of the lowest and the highest notes. From the construction of the Western tonal music scale, we have:

$$\frac{f_{k+1}}{f_k} = 2^{\frac{1}{12}} or\ f_{k+1} \approx 1.059 f_k \tag{3.8}$$

For instance, a C1 note has a frequency of $f_{C1} = 32.7$Hz; the frequency of the next note C#1 is: $f_{C\#1} = 32.7 + 0.06 * 32.7 = 34.6$Hz.

It is unlikely that two adjacent low-frequency notes will be played simultaneously in Western tonal music, because this is generally unpleasant for the ears. However, chromatic notes that are played successively often interfere in time and may be superimposed during a lapse of time. This is for instance the case in most of Chopin piano music where the extensive use of the sustain pedal results in mixtures of adjacent low-frequency notes. The constraint regarding the minimum length of the analysis window needs thus to be taken into account for low pitches.

The frequency components of the DFT are equally spaced and thus have a constant frequency resolution. To discriminate adjacent pitches, particularly at low frequencies, a sufficiently long window length is thus required whereas it is unnecessary when considering higher pitches. Multi-resolution approaches have been proposed as an alternative to the conventional linear frequency and constant resolution of the DFT. In this dissertation, we focus on a multi-resolution approach commonly used in music audio analysis: the Constant-Q transform (CQT). This representation has been used for chromagram computation in many works related to chord or key estimation.

3.2.4.3 Constant-Q Transform

One common approach to solve the time/frequency resolution dilemma is to perform a frequency-varying multi-resolution analysis. In this case, the frequency spectrum is split into subbands and each one is processed independently from the others. This allows the use of shorter analysis windows at higher frequencies while lower frequencies can still have the required frequency resolution to separate closely spaced sinusoids. An interesting approach was presented in 1991 by Brown [Bro91] who proposed to use the constant-Q transform for music signal analysis. The constant-Q transform is a spectral analysis where frequency domain channels are not linearly spaced, as in DFT-based analysis, but geometrically spaced (the center frequency to resolution ratio $Q = \frac{f}{\Delta f}$ remains constant), thus tightly similar to the frequency resolution of the human ear. The CQT transform is closely related to the Fourier transform but gives a better representation of spectral data from a music signal. The center frequencies that are distributed geometrically follow the equal tempered scale used in Western music. Note that the CQT was introduced earlier, outside the musical context, see for instance [YB78].

In case of musical applications, the calculation of the CQT is based on the frequencies of the equal tempered scale. The constant Q transform of a discrete signal $x(n)$ can be calculated as:

$$X^{cq}(k) = \sum_{n=0}^{N(k)-1} w(n,k) x(n) e^{-j2\pi f_k n} \tag{3.9}$$

where $X^{cq}(k)$ is the k^{th} component of the constant-Q transform. For each value of k, the window function $w(n,k)$ varies proportionally to the center frequency f_k. Let Q denote the constant ratio of frequency to resolution, $Q = \frac{f_k}{\Delta f_k}$, and let f_s denote the sampling rate. The length of the window $w(n,k)$ in samples at frequency f_k is $N(k) = \frac{Q.f_s}{f_k}$. $N(k)$ depends on the frequency and thus on the bin position k.

Figure 3.3 represents the window length N (in seconds) with respect to the frequency (in Hertz), for a $\frac{1}{2}$-tone spacing ($Q = (2^{\frac{1}{12}} - 1)$). For instance, a window of 0.5s (duration of a beat at a tempo of 120 bpm) corresponds to a frequency value of 104Hz. The constant-Q transform increases time resolution towards higher frequencies. The length of the window $w(n,k)$ decreases with frequency.

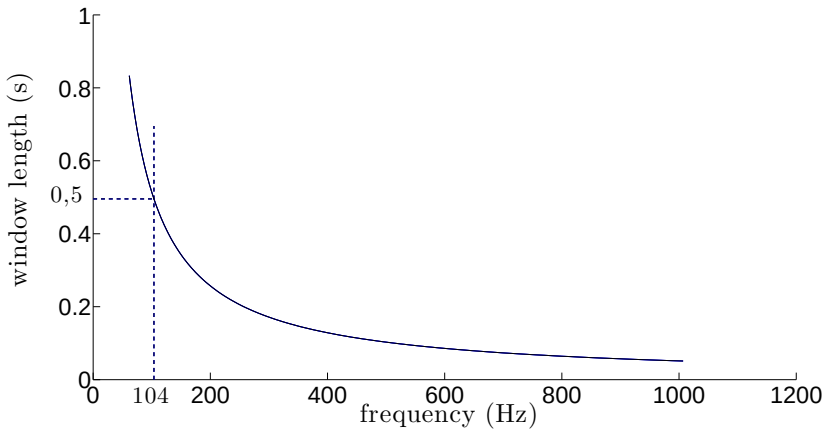

Figure 3.3: Length of the constant Q window in seconds with respect to the frequency in Hertz.

[BP92] proposes an efficient algorithm of the CQT that takes advantage of the Fast Fourier Transform (FFT) so that the computation cost are reduced as compared to the direct evaluation of the CQT.

3.3 Chroma Representation, Background

Because they are a powerful compact representation of the tonal content information of the signal, chroma features have been widely used as input features of music analysis models based on the music harmonic content, such as chord or key finding, cover song detection or structure estimation. Various approaches for chroma computation exist. Although they present some variances in the implementation, they follow in general the same guideline that consists of two main steps:

1. First, a semitone pitch class spectrum (SPS), that is a log-frequency representation of the spectral content of the music audio signal, is constructed. It is expressed in a MIDI-note scale and is either computed from the Fourier transform or from the constant-Q transform. The center frequencies of the CQT can be chosen according to

the frequencies of the equal-tempered scale. In such a case, the constant-Q spectrum corresponds to a semitone pitch class spectrum.

2. Secondly, the semitone pitch spectrum is mapped to the chroma vectors. For this, the semitones in octave distance are added up to pitch classes.

The chromagram computation may include some other steps such as a pre-processing step that separates harmonic and noise components, a filtering step that smoothes the chromagram or a post-processing normalization step that makes the chromagram invariant to dynamics. We review in the following some chroma feature extraction methods.

3.3.1 Chromagram Based on the Fourier Transform

In many approaches, the chromagram is generated using the Fourier transform. This approach was first proposed by Fujishima in [Fuj99], where the input signal is transformed from the time to the frequency domain using an FFT. Frequency bins corresponding to a same semitone are summed up to form a semitone pitch spectrum, which is then folded to pitch classes, resulting in a PCP vector.

This approach was followed by a large number of researchers with some variants. In some approaches, the resolution of the chromagram is increased in order to improve robustness against tuning and other frequency oscillations, such as in the work of Goto [Got06], where a chromagram is computed so that there are 100 cents to a tempered semitone. Some approaches introduce a filtering process to reduce transient and noise, such as in the work of Peeters [Pee06b].

The FFT is particularly blurred at low frequencies. In order to identify strong tonal components in the spectrum and to get a higher resolution estimate of the underlying frequency, Ellis & Poliner [EP07] do not compute the chroma feature directly from the FFT. They use the Instantaneous Frequency spectrum, which uses the phase derivative to interpolate the frequency distribution.

3.3.2 Considering the Harmonics in the Pitch Class Profiles

Some methods for chroma computation take into account the higher harmonics of the notes in the chroma features computation. For instance Gómez introduces in [GÓ6a] an extension of the PCP, the Harmonic Pitch Class Profiles (HPCPs). A weighting procedure is proposed in order to make harmonics contribute to the pitch class of its fundamental frequency, so that each peak frequency f_i has a contribution to the frequencies having f_i as harmonic frequency (f_i , $\frac{f_i}{2}$, $\frac{f_i}{3}$, $\frac{f_i}{4}$, ...).

Lee [Lee06a] proposes a feature vector called the Enhanced Pitch Class Profile (EPCP) for the application of chord recognition from audio. The chromagram is computed from the Harmonic Product Spectrum (HPS) instead of the DFT. The use of a HPS allows the elimination of non-tonal signal components from the spectrum.

Pauws [Pau04] computes the chromagram using an auditory perception inspired front-end so that the perceptual pitch and the musical background are simultaneously taken into account.

3.3.3 Constant-Q Profiles

Some approaches derive the chromagram from a CQT instead of the FFT. In [PBO01], Purwins *et al.* propose to compute CQ-profiles that are 12–dimensional vectors similar to chroma vectors. The CQT filters are chosen so that they correspond to musical notes. The Constant-Q spectrum thus directly corresponds to a semitone pitch spectrum from which 12-dimensional vectors (corresponding to the 12 pitch classes) can be computed.

This approach has been often adopted by other researchers, possibly with some variations. For instance, Harte & Sandler [HS05] propose a tuning algorithm for a CQ-based chromagram. In [BP05], Bello & Pickens generate the chromagram using a constant-Q transform. A resolution of 36 bins per octave is used. The chromagram is low-pass filtered to eliminate sharp edges.

3.3.4 Chromagram Based on multi-f0s

Some approaches compute chroma features from a multi-pitch representation instead of a spectral representation. For instance, Ryynänen & Klapuri [RK08b] compute a chromagram from a pitch salience estimator. In [ZR07], Zenz & Rauber compute a multi-pitch based chromagram using the Enhanced Autocorrelation (EAC) algorithm described by Tolonen et al [TK00]. Varewyck *et al.* [VPM08] also propose a chroma extraction method based on multiple pitch tracking techniques.

3.3.5 Filter bank

In the context of audio matching, Müller *et al.* [MKC05] introduce a new kind of chroma-based audio feature referred to as CENS features (Chroma Energy distribution Normalized Statistics) that presents a high degree of robustness to variations in parameters such as dynamics, timbre, articulation and local tempo deviations. In this approach, the chroma features are computed by the use of a filterbank with fixed frequency bands. The audio signal is decomposed into subbands corresponding to notes A0 to C8 (MIDI pitches 21 to 108). The short-time mean-square power is computed over each subband using a 200ms with an overlap of half the size. The chroma vectors are obtained by adding up the corresponding short-time mean-square power (STMSPs) of all pitches belonging to the 12 respective pitch class. The chroma vectors are normalized to be invariant to dynamic variations and then quantized by applying energy thresholds in order to be insensitive to noise components. In order to smooth local tempo deviations and slight variations in note groups, such as trills or arpeggios, a much larger statistics window is then considered.

3.3.6 Extension: the Tonal Centroid

Another feature devoted to harmonic analysis has recently been proposed. We give here a brief overview of this feature since some recent works [LS08] [LB07] have shown that it may be a powerful feature for harmonic analysis.

The *Tonal Centroid* was introduced in [HSG06] as a new feature for detecting changes in the harmonic content of musical audio signals. A harmonic Centroid transform is applied to the chromagram decomposition so that the 12-dimensional chroma vectors are mapped to a six-dimensional Hypertorus structure. The Tonal Centroid is derived from an old planar representation of pitch relations called the Harmonic Network or *Tonnetz*. In this representation, close harmonic relations such as fifths and thirds appear as small Euclidian distances on the plane. Three circularities are considered: the circle of fifths, the circle of minor thirds and the circle of major thirds.

When enharmonic equivalence (C# equivalent to Db) and octave equivalence (C1 equivalent to C2) are assumed, the Tonnetz, which is theoretically an infinite plane, can be wrapped into a tube with the line of fifths becoming a helix on its surface. In the Spiral Array model [Che02], the two ends of the tube are joint together, resulting into a hypertorus with the circle of fifths wrapping around its surface three times. The Tonal Centroid is a 6-dimensional interior space contained by the surface of the Hypertorus. The 6 dimensions can be visualized as a projection onto the circle of fifths, the circle of minor thirds and the circle of major thirds and represented as three coordinate pairs $(x1, y1)$, $(x2, y2)$ and $(x3, y3)$ (see Figure 3.4).

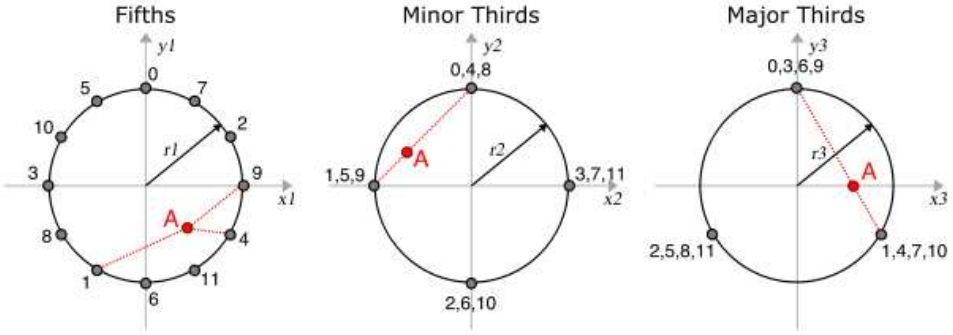

Figure 3.4: Graphical representation of the 6-dimensional Tonal Space as three circles. From left to right: circle of fifths, circle of minor thirds and circle of major thirds. The Tonal Centroid for chord A Major (pitch classes 9, 1 and 4) is shown at point A. Adapted from [HSG06].

Let c denote a 12-dimensional chroma vector and let Φ denote the transformation matrix that represents the basis of the 6-dimensional space.

$$\Phi = [\Phi_1, \Phi_2, \Phi_3, \Phi_4, \Phi_5, \Phi_6] \tag{3.10}$$

where

$$\phi_l = \begin{bmatrix} \Phi(1,l) \\ \Phi(2,l) \\ \Phi(3,l) \\ \Phi(4,l) \\ \Phi(5,l) \\ \Phi(6,l) \end{bmatrix} = \begin{bmatrix} r_1 sinl\frac{7\pi}{6} \\ r_1 cosl\frac{7\pi}{6} \\ r_2 sinl\frac{3\pi}{2} \\ r_2 cosl\frac{3\pi}{2} \\ r_3 sinl\frac{2\pi}{3} \\ r_3 cosl\frac{2\pi}{3} \end{bmatrix} \tag{3.11}$$

where the values r_1, r_2 and r_3 are the radii of the three circles. In [HSG06], they are set to 1, 1 and 0.5 respectively, ensuring that the distances between pitch classes in the 6-dimensional space correspond to our perception of harmonic relations between pitches. The 6-dimensional Tonal Centroid vector ξ is obtained from the 12-dimensional chroma vector c according to the following equation:

$$\xi(d) = \frac{1}{||c||_1} \sum_{l=1}^{12} \phi(d,l)c(l) \quad \begin{matrix} 1 \leq d \leq 6 \\ 1 \leq l \leq 12 \end{matrix} \tag{3.12}$$

3.3.7 Why Using Chroma Features for Harmonic Content Analysis?

We have chosen to use the chroma representation because we think that it is a very intuitive and natural representation of the signal in terms of harmony. We find it particularly convenient for chord analysis: the 12 bins of the chroma features correspond to the traditional pitch classes of the equal tempered scale. The chromagram can be followed as a music score when listening to the music.

3.4 Derivation of Chroma Features

In what follows, we focus on the derivation of three chroma representation extraction methods. The first two are based on the two above-mentioned spectral representations of the signal (FFT and CQT), the third one is based on a multipitch tracking technique. These approaches will be analyzed and compared in Section 3.6.

3.4.1 Chroma Based on a Spectral Representation

We review here two chromagram computation methods based on a spectral representation. The first one is based on the conventional fixed resolution FFT and the second one is based on the multi-resolution CQT. The two methods follow the same general schema represented in Figure 3.5. We start by estimating the tuning of the piece. The chromagram is computed in three steps after tuning estimation. First, the values of the DFT/CQT are mapped to a semitone pitch spectrum. The corresponding channels are then smoothed over time. Finally, the resulting semitone pitch spectrum is mapped to the semi-tone pitch classes.

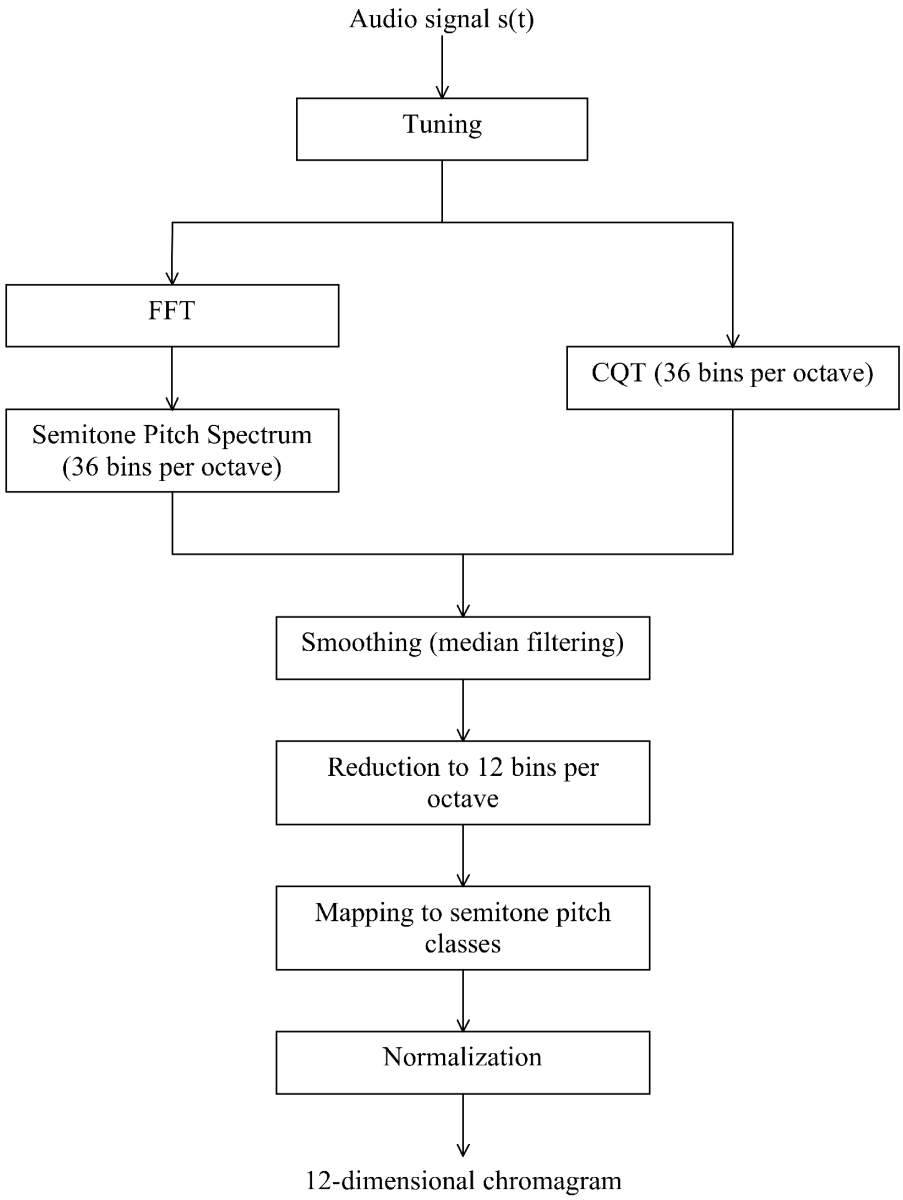

Figure 3.5: General flowchart of chromagram computation.

3.4.1.1 Tuning

The chroma values are obtained by mapping frequency values of the spectrogram to the semitone pitch classes that are based on a standard reference frequency $T_{ref} = 440$Hz. The energy peaks in the spectrogram will be mapped to the chroma vectors. It is therefore important that the peak frequencies correspond as close as possible to usual pitch values (262.6, 277.2, 293.7, ... Hz). Since the instruments may have been tuned according to a reference pitch different from the standard $A4 = 440$Hz , it is necessary to estimate the tuning of the track. After computing the precise tuning used in a given song, we center

the semitone pitch spectrum filters accordingly so that they fall precisely in the middle of a note. We now detail the tuning estimation method we use in this work.

In our approach, we estimate the reference frequency (or tuning) before mapping frequency values to the pitch classes. Other works propose to first compute a chromagram and then tune it according to a determined reference frequency. For instance [HS05] or [BP05] built a 36 bins-per octave resolution chromagram. They then compute the histogram of the chromagram peaks distribution across one semitone width (corresponding to 3 bins). The maximum point in the histogram gives the semitone centre tuning value.

Here, the tuning is estimated using the method proposed by Peeters in [Pee06b]. We assume that the tuning is constant over the music track duration. The amount of energy of the spectrum explained by the frequencies corresponding to the semitones based on each candidate tuning is measured. The candidate tuning that allows us to explain the best the energy of the spectrum is selected as the tuning of the track.

Let us consider a set of tuning candidates between 427Hz and 452Hz, which correspond to the quartertones below and above A4. The candidate tunings are successively tested as following. For a given tuning test t and a given signal frame m, we define the *modeling error* $\epsilon(t, m)$ as the ratio between the energy of the spectrum explained by the current tuning t and the total energy of the spectrum.

$$\epsilon(t, m) = 1 - \frac{\sum_n A(f_{t,n}, m)}{\sum_f A(f, m)} \tag{3.13}$$

where A denotes the amplitude of the Fourier transform and $f_{t,n}$ are the frequencies of the semi-tones pitches n (in MIDI) based on the tuning t:

$$f_{t,n} = t \cdot 2^{\frac{n-69}{12}}, t \in [427, \ldots, 452] \tag{3.14}$$

The energy of the current tuning t is computed as the sum of the energy at the frequencies f_t corresponding to the semi-tones pitches based on the tuning t. A low value of ϵ indicates that most of the peaks of the spectrum correspond to notes based on the tested tuning. The estimated tuning is chosen as the value t that minimizes the modeling error over time. The estimated tuning T_{ref} is taken into account when computing the chromagram, as explained below.

In practice, many audio files are not based on a tuning of $A4 = 440$Hz . As an illustration, we represent in Figure 3.6 the histogram of the tunings estimated over the widely used *Beatles test-set* for chord recognition (see Chapters 2, Section 2.3.2). It shows that most of the songs are not based on a tuning of $A4 = 440$ Hz. The estimated tunings of the tracks are comprised between 430 Hz and 444 Hz.

Note that we assume here a constant tuning over the whole duration of the piece. To reduce the computation cost, it would be possible to compute the tuning of the piece on a short extract (using only 30s of music for instance).

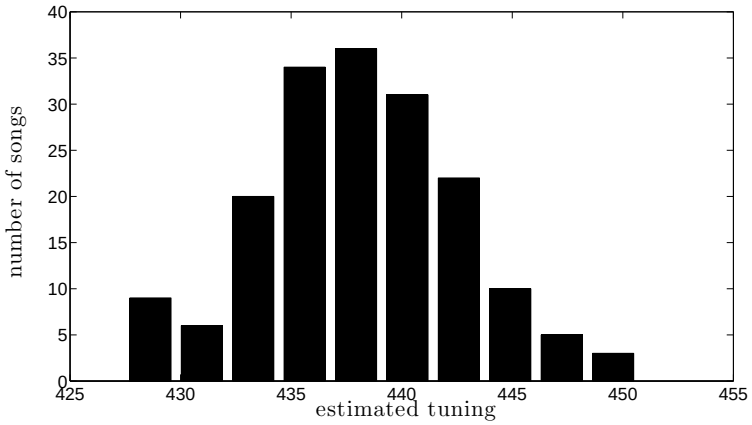

Figure 3.6: Histogram of the estimated tunings over the *Beatles test-set*.

3.4.1.2 Frequency Region Selection for Chroma Computation

The chroma vector is obtained by converting the signal into the frequency domain, using an FFT or a CQT, and mapping the calculated intensities in the frequency bins corresponding to the music pitches. In the mapping, we do not consider all the frequencies of the spectrum $X(k)$ but the analysis is restricted in general to a frequency region that corresponds to the most relevant frequency values for pitch distribution.

Let f_{min} and f_{max} respectively denote the minimum and maximum frequencies of $X(f)$ considered in the chroma computation. In what follows, and n_{min} and n_{max} denote the midi notes corresponding to f_{min} and f_{max}. Assuming a tuning of $A4 = 440$Hz, a midi note n is related to its frequency f by the following equation:

$$n = 12log_2\frac{f}{440} + 69 \tag{3.15}$$

The various works that use chroma features as a representation of the harmonic content of the signal do not consider the same frequency region for chromagram computation. For instance, Bello & Pickens [BP05] compute the chromagram from 98Hz to 5250Hz whereas Oudre *et al.* [OGF09b] limit the frequency range to the interval $73.42-587.36$Hz, although both of them are used as input of a chord estimation algorithm evaluated on the *Beatles test-set*.

The selection of the frequency region depends on many criteria. Different frequency regions should be selected depending on the possible presence of noise or percussive sounds in the signal, depending on whether the chroma extraction algorithm considers the presence of higher harmonics or not, or depending on the instrumentation.

In our work, because of frequency resolution limits (the frequency distance between adjacent semitone pitches becomes small in low frequencies), we only consider frequencies above $f_{min} = 60$Hz.

In our experiments on the Beatles test-set, we found that the best results were obtained

considering only the frequencies below $f_{max} = 1000$Hz. The upper limit is set to 1kHz because the fundamentals and partials of the music notes in popular music are usually stronger than the non-harmonic components up to 1kHz [Mad06]. This is illustrated in Figure 3.7, in Section 3.5.1 of this Chapter. Our choice for f_{max} is also supported by the fact that many of the higher partials, which are whole number multiples of the fundamental frequency, are far from any note of the Western chromatic scale. This is especially true for the 7^{th} and the 11^{th} partials. We found that the best results on a classical music test-set, the *Mozart piano test-set* are obtained using components up to 2kHz. Further experiments should be devoted to better study the influence of the frequency region selection parameter.

Note that some approaches such as [MND09] or [RK08b] use jointly several chromagrams instead of one, in order to distinguish between various registers. Each chromagram is computed considering a different frequency region that may capture for instance to the bass or the melody content.

3.4.1.3 Computation of a Semitone Pitch Spectrum

Semitone Pitch Spectrum from the FFT

We review here a method that was initially proposed in [Pee06a]. In our analysis, the signal is down-sampled to $11025Hz$, converted to mono and converted to the frequency domain by a FFT using a Blackman window of length N with 12.5% overlap. The value of N will be discussed further, in part 3.6.4. The values of the FFT are mapped to a semitone pitch spectrum according to the estimated tuning using the mapping function:

$$n(f_k) = 12\log_2(\frac{f_k}{T_{ref}}) + 69, n \in \Re^+ \tag{3.16}$$

where f_k are the frequencies of the notes in the Fourier transform and n corresponds to the semitone pitch scale values expressed in a MIDI-note scale. For each MIDI note of frequency f_k of the semitone pitch spectrum, we consider the frequencies of the spectrum that are contained in a window centered around f_k. The contribution of the peaks of the DFT bins comprised in the considered window is weighted using a set of filters described below.

Let us define a set of filters $H_{n'}$ centered on the semi-tone pitch frequencies $n' \in [n_{min}, n_{min}+1, \ldots, n_{max}]$. For instance, if we consider the notes comprised between the frequencies 60Hz and 1000Hz (B1 to B5), the filters will be centered on the MIDI notes $n' \in [35, 36, \ldots, 83]$. Frequency resolution is a salient parameter in pitch class features computation. Chroma features are in general represented as 12-dimensional vectors that correspond to the 12 semitones of the equal tempered scale. Nevertheless, it may be pertinent to increase the semitone resolution to improve robustness against tuning and other frequency variations, such as the vibrato of an instrument. In this case, a semitone is represented by several filters instead of one (typically 2 or 3). In order to increase the semitone resolution, we define a factor $R \in \Re^+$ that sets the number of filters used to represent one semitone. The center of the filters are now set on the MIDI notes $n' \in [n_{min}, n_{min}+\frac{1}{R}, n_{min}+\frac{2}{R}, \ldots, n_{max}]$.

Each filter can be defined by the function

$$H_{n'}(f_k) = \frac{1}{2} tanh(\pi(1-2x)) + \frac{1}{2} \tag{3.17}$$

where x is the relative distance between the center of the filter n' and the frequencies of the Fourier transform: $x = R|n' - n(f_k)|$. The filters are equally spaced and symmetric in the logarithmic semitone pitch scale, extend from $n'-1$ to $n'+1$ with a maximum value at n'. The values of the semi-tone pitch spectrum $N_{FFT}(n')$ are then obtained by multiplying the Fourier transform values $A(f_k)$ by the set of filters $H_{n'}$:

$$N_{FFT}(n') = \sum_{f_k} H_{n'}(f_k) A(f_k) \tag{3.18}$$

Semitone Pitch Spectrum from the CQT

The CQT is closely related to a semitone pitch spectrum. Let β denote the number of bins of the CQT per octave. Chroma features are usually represented in a 12-bin histogram, each bin corresponding to one of the 12 semitones of the equal-tempered scale. In the case of $\beta = 12$ (semitone spacing), the center frequencies directly correspond to musical notes of the semitone pitch scale and the computation of the constant Q transform leads to a semitone pitch spectrum $N_{CQT}(n')$. Very often, as in the case of the FFT-based chroma feature computation, a higher resolution is used to get a finer pitch class representation. We use here a 36-bins per octave resolution. When $\beta = 36$, each note in the octave is mapped to 3 bins in the chroma and the computed CQT spectrum corresponds to a $\frac{1}{6}$-tone pitch spectrum.

Let $f_{min,440}$ be the minimum frequency considered in the signal feature computation in the ideal case of a perfect tuning. The actual minimum frequency value f_{min} is chosen according to the estimated tuning of the track: $f_{min} = f_{min,440} * \frac{T_{ref}}{440}$. The center frequencies are geometrically spaced, according to the frequencies of the equal-tempered scale:

$$f_k = (2^{1/\beta})^k f_{min} \tag{3.19}$$

As stated in Equation (3.9), the CQT time resolution increases towards higher frequencies. The length of the analysis window decreases with the frequency. Here, the hopsize is chosen to be equal to the smallest window length.

3.4.1.4 Smoothing

Transient reduction can be done during chromagram computation, as proposed in [Pee06a], the semitone pitch spectrum $N_{FFT}(n')$ or $N_{CQT}(n')$, simply denoted from now by $N(n')$ is computed for each frame m and is then smoothed over time using a median filtering. This provides a reduction of transients and noise. Note that smoothing of the semitone pitch spectrogram strengthens spectral envelope continuity, a physical property; while

smoothing on the chromagram does not rely on any physical property. This is why the filtering is performed on the notes rather than on the chroma vectors.

In general, when increasing resolution, only one filter per semitone is considered so that the final chroma feature is 12-dimensional and can easily be compared with chord or key profiles[2]. In our chroma feature implementation based on the FFT, for each semitone $n' \in [n_{min}, n_{min}+1, \ldots, n_{max}]$, we select the filter centered on the exact pitch. For instance, for a 36-bins per octave resolution, we only consider the filter centered on $n' = 69$ for the A4 note, not the ones centered on $n' = 68.666$ and $n' = 69.333$. This can be done because the tuning is now guaranteed to be 440 Hz. This process also provides a reduction of the influence of noise in the computation of the chroma features.

3.4.1.5 Chroma Spectrum

The mapping between the semitone pitches n and the semitone pitch classes (chroma) c is defined as:

$$c(n) = \mod(n, 12) \tag{3.20}$$

All the semitones pitches corresponding to equivalent pitch classes are added so that we obtain a sequence of 12-dimensional chroma feature vectors. Each of the 12 bins l of the chroma vector can be calculated from the semitone pitch spectrum $N(n')$as:

$$C(l) = \sum_{n' \ so \ that \ c(n')=l} N(n'), \ l \in [1, 12] \tag{3.21}$$

3.4.1.6 Post-processing: Normalization

The chromagram is in general normalized to provide robustness against variations of dynamics. This normalization post-processing step can be done so that the components of each chroma vector sums to unity, as we do here. This choice is followed in many other works [LB07] [RK08b]. Other methods propose to normalize the chromagram for each frame by its maximum value [GÓ6b] [CC05a].

3.4.2 Chroma Based on multiple f0s

In the last few years, the problem of estimating the fundamental frequency, or $f0$, of the signal is a task that has attracted the attention of a growing number of researchers. This is because it is an extremely important descriptor of the signal. It is largely admitted that $f0$ estimation is equivalent to pitch estimation. In the case of polyphonic music, several musical notes are played simultaneously and the term *multiple-f0* is used. The *multiple-f0* algorithms allow retrieving the various pitches that have been produced.

The idea of deriving a chroma representation from the output of a multiple pitch tracking technique comes out naturally. It has been already explored in [RK08b] and

[2] A key/chord profile is a 12-dimensional template that indicates the perceptual importance of each note of the equal-tempered scale within a key or a chord. More details will be given in the next chapters.

[VPM08] for instance. Here, we are interested in comparing the approaches based on spectral representation with an approach based on a multiple pitch tracking technique for chroma features computation. For this, we rely on a multiple-f0 estimation algorithm proposed by Yeh in [Yeh08] and [YRC08]. We thank C. Yeh for providing his code.

Briefly, to estimate the pitches of the notes in the audio signal, we use the frame-based $f0$ estimation algorithm proposed in [Yeh08]. It is based on a score function which evaluates the plausibility of a set of $f0$ hypotheses. It works in four stages.

1. First, an adaptive noise level is estimated in order to classify the spectral peaks into sinusoids (above the noise level) and noise (below the noise level).
2. Secondly, a set of $f0$ candidates is iteratively extracted until all the significant sinusoidal components are explained.
3. Thirdly, a score function jointly evaluates all the combinations of $f0$ candidates. It is based on four criteria: harmonicity (harmonic matching that estimates the partial frequencies and amplitudes of the hypothetical sources), mean bandwidth (envelope smoothness), spectral centroid (energy concentration in lower partials) and synchronicity (synchronous amplitude evolution within a single source).
4. Finally, the best combination of $f0$ candidates is selected by a polyphony inference algorithm.

The output of the multiple-f0 estimation algorithm can be seen as a semitone pitch spectrum: for each frame, it gives an estimation of the pitch and salience of the notes present in the signal. This semitone pitch spectrum covers several octaves. It is reduced to one octave by adding each pitch's intensity to the pitch class of its chroma. The resulting feature is a 12-dimensional chroma vector.

3.5 Two Problems Related to the Chroma Features

3.5.1 Chroma Features and Harmonics

Let us consider a chroma feature extraction method based on a spectral representation (FFT or CQT). As explained in part 3.2.3, a note generated by an instrument produces a set of harmonics. In a spectral representation, we do not directly observe the various pitches but a mixture of their harmonics that will result in a mixture of non-zero values in the chroma vector. It is thus important to note that the chroma vector of a note played by an instrument does not only contain the pitch classes corresponding to the fundamental frequency $f0$ of the perceived pitch p_0 (ignoring octave considerations) but also include a mixture of their harmonics.

Figure 3.7 shows a chroma feature of a cello C1 note ($65,4$Hz) considering various frequency intervals (from $f_{min} = 60$Hz to various values of f_{max}) for computation. We can follow the apparition of the harmonics of the C: C-C-G-C-E-G and so on, as well as of some other components related to the residual part expressed in Equation (3.3), especially when

high frequencies are considered in the feature computation. The problem of harmonics in the chroma features will be further discussed in Chapter 4 of this dissertation.

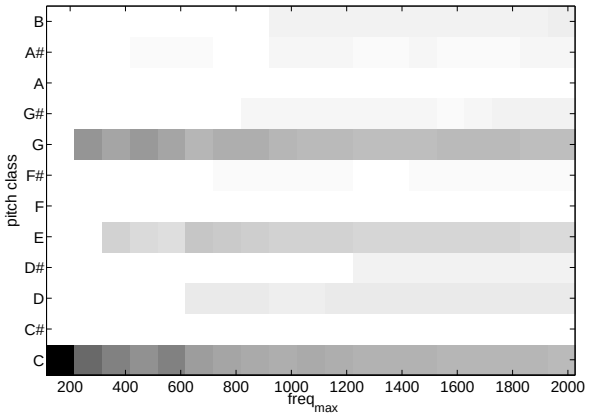

Figure 3.7: Chroma feature of a cello C1 note considering various frequency intervals (from $f_{min} = 60$Hz to $f_{max} \in [100, 2000]$ Hz) for computation.

3.5.2 Beat-Synchronous Analysis

Most of the works that extract harmonic content information from audio signals rely on chroma features. In some cases, it can be very useful and even necessary to perform a beat-synchronous analysis, that is to compute one feature per beat. The computation of beat-synchronous chroma features has thus become quite common in harmonic content analysis models. Beat-synchronous chroma features have been used in many approaches that attempt to estimate the chord progression of an audio file, as for instance [BP05], [RSN08], [SIY+08], [YKK+04], [ZR07]. [BP05] argue that beat-synchronous analysis frames help to overcome noise introduced by transient components in the sound (drums and guitar strumming) and short ornamentations, thus minimizing the effect of local variations. The use of beat-synchronous chroma features is convenient in music similarity and cover song identification tasks [Mad06] [MKL06] [SW05] [BW01] [BW05], especially when comparing the chord progression of two songs, possibly at different tempo. Indeed, this provides invariance to tempo changes [Ell06] [EP07]. Beat-synchronous chroma features may be useful for music segmentation and music structure detection, in particular in approaches that combine harmonic and metrical information and need to work with features related to the meter [PP08b] [Mad06].

In this section, we wish to underline several issues related to the use of beat-synchronous chroma features. We shall conduct in Section 3.6.3 several experiments that illustrate our purpose. In Chapter 5, we shall propose a model that takes into account interaction between chords and downbeats. The proposed model requires features related to the meter. We will use one single input vector per beat/tactus (or per half-beat/tatum).

3.5.2.1 Towards a Beat-Synchronous Analysis

Beat-synchronous chroma features can be obtained in various manners.

In the case of a fixed resolution analysis (using a FFT), beat-synchronous chroma vectors can be obtained from the frame-by-frame analysis in two ways.

1. We can either compute a frame-by-frame chromagram using a fix length of analysis frame and then averaging the chromagram according to the tactus/tatum positions (see Figure3.8, top). In what follows, we will refer to this approach as a *beat-average analysis* denoted by B_{AV}. This approach is adopted in [EP07].

2. Or we can perform a beat-synchronous analysis by using an adaptive window length that is defined by the beat positions (see Figure 3.8, bottom). In this case, each analysis frame corresponds to a beat and there is no overlap between successive frames. In what follows, we will refer to this approach as a *beat-adaptive analysis* denoted by B_{AD}. This approach is adopted for instance in [ZR07].

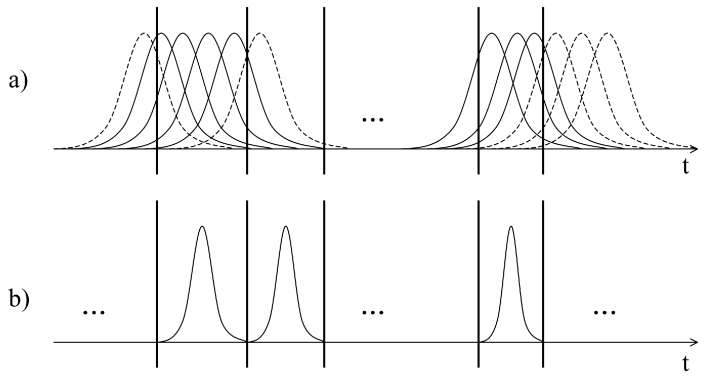

Figure 3.8: Two FFT-based beat-synchronous analysis: a) B_{AV}, b) B_{AD}. Dashed lines correspond to the frames that are not taken into account in the computation of the beat-synchronous chroma vector.

In the case of multi-resolution analysis, the length of the window is determined by the frequency. The beat-synchronous chroma features can thus only be obtained by averaging frames according to beat locations. This approach is adopted in [BP05].

The various investigated methods for chroma features computation are listed in Table 3.3.

A more detailed discussion about beat-synchronous chroma features with quantitative evaluation follows in part 3.6.3.

3.5.2.2 Problem of Mixing Harmonies

Let us consider a simple case of chord progression with one chord change per beat. Let $c_1, c_2, c_3, \ldots$ denote the successive chroma features computed on overlapping frames. Let

Table 3.3: Summary of the investigated methods for computing the chroma features. FFT_L/FFT_S: frame-by frame FFT-based method using a long (0.5s)/short(0.125s) analysis window, CQT: CQT-based method, B_{AV}: *beat averaged analysis*, B_{AD}: *beat adaptive analysis*.

	FFT	CQT
Frame-by-frame	FFT_L 0.5s FFT_S 0.125s	CQT
Beat-synchronous	$FFT_L B_{AV}$ $FFT_S B_{AV}$ $FFT B_{AD}$	$CQT\ B_{AV}$

b_k and b_{k+1} denote two successive beat positions and N_k denote the number of overlapping chroma vectors that are comprised between b_k and b_{k+1}.

A common approach used to obtain a beat-synchronous chroma feature C_k is to compute the average of the N_k overlapping frames that are comprised between two considered beat positions b_k and b_{k+1} (see for instance [BP05], [PP08b], [Ser07]):

$$C_k = \frac{1}{N_k} \sum_{b_k \leq n < b_{k+1}} c_n \tag{3.22}$$

This is illustrated in Figure 3.9. This method will be referred to as $B_{AV\,mean}$ method or as B_{AV} method when there is no ambiguity.

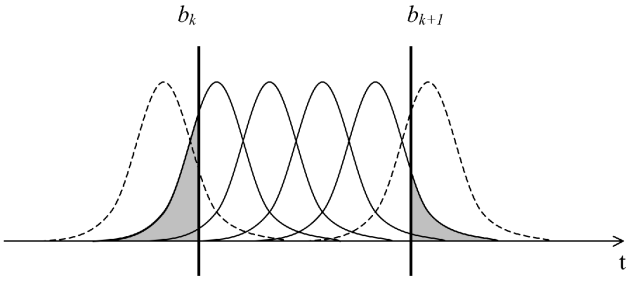

Figure 3.9: Computation of a beat-related feature by averaging overlapping frames between two successive beat positions b_k and b_{k+1}. Solid lines correspond to the frames that are taken into account in the computation of the beat-related chroma feature. The grey areas correspond to information related to harmony that does not correspond to the considered chord.

Another possible way of computing beat-synchronous chroma features from frame-based features is to take the median (in the time direction) over all the chroma frames falling between two consecutive beat positions [MND09]. This will be referred to as the $B_{AV\,median}$ method. We will compare the two possibilities in part 3.6.3.

Ideally, the beat-synchronous chroma features should capture the harmonic content of each single chord. However, some spectral information that comes from adjacent chords is mixed with the spectral information of the considered chord, as represented in Figure

3.9. The amount of spectral information coming from the adjacent chords increases with the length of the analysis window. It would be thus desired to use small window lengths. This is in conflict with the need of sufficiently large windows for resolution considerations. We thus need to make a trade-off between considering low pitch frequencies and mixing spectral information between adjacent chords.

Let us illustrate this on an example: Chopin's Study op. 25 no 10 (*Octaves*). The opening of this study, represented in Figure 3.10, consists of a series of eight-note-tuplets octaves in cut time, played at a very fast tempo, *Allegro*. There is one chord per eight-note and each chord corresponds to a single note played at four different octaves (in practice adjacent chords may mixed up because of the use of the pedal).

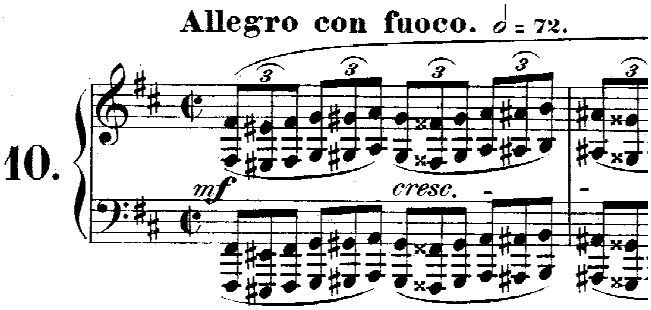

Figure 3.10: Opening of the Chopin Study *Octaves*.

Figure 3.11 represents three variations of the FFT-based chromagram computed on the considered music excerpt and averaged on the eighth notes.

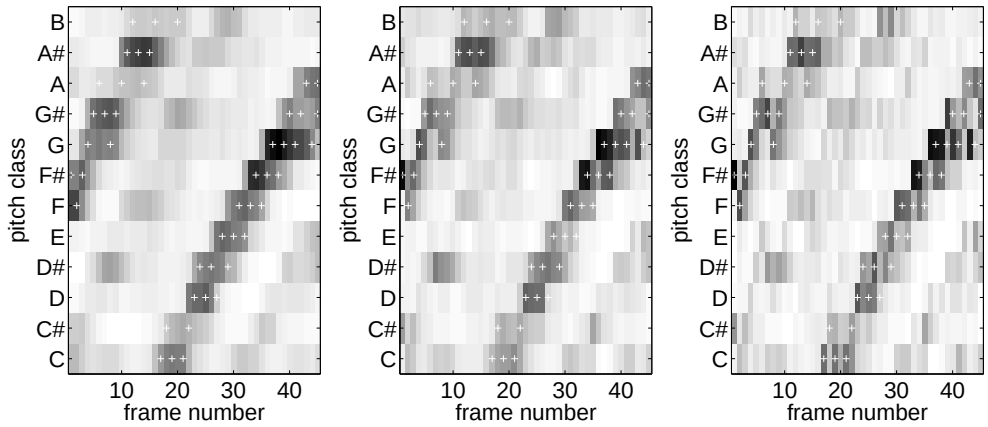

Figure 3.11: FFT-based chromagram computed on the beginning of the Chopin Study *Octaves*. From left to right: FFT_LB_{AV}, FFT_SB_{AV}, $FFTB_{AD}$. The blank"+" signs represent the successive notes played at four different octaves, according to Figure 3.10.

It can be seen that the chord transitions are much clearer in the case of *beat-adaptive analysis* than in case of *beat-average analysis*. However, if we look at the semitone pitch spectrum, Figure 3.12, we can distinguish chromatic scales at 4 different octaves in case

FFT_LB_{AV} whereas some notes in the low frequencies are not correctly detected in cases $FFTB_{AD}$ and FFT_SB_{AV}. This is because the analysis is done using a window that is too short regarding the frequency resolution that is needed. A longer analysis window would be required to detect precisely the low octaves notes.

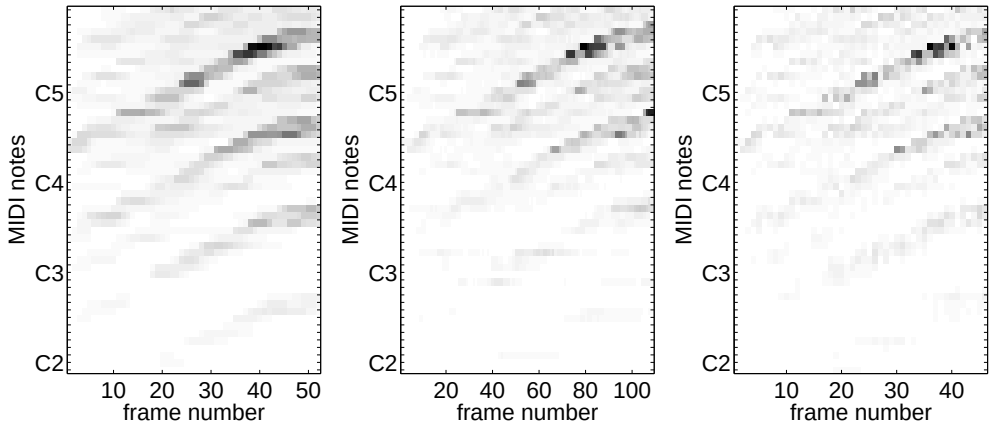

Figure 3.12: FFT-based semitone pitch spectrum computed on the beginning of the Chopin Study *Octaves*. From left to right: FFT_LB_{AV}, FFT_SB_{AV}, $FFTB_{AD}$.

The problem of mixing the harmonic content of two different chords in one beat-synchronous chroma vector occurs at the point where the harmony changes. The Chopin's study example is an extreme case. In general the harmony of a piece changes much slower, especially in popular music where very often the chords have duration of a measure or half a measure (even if this is not a rule). On the one hand, the longer the length of the analysis window is, the more undesirable harmonic information from the adjacent chords the beat-synchronous chroma feature will capture. On the other hand, a sufficiently long window is required to detect precisely the low pitches notes. This trade-off should be kept in mind when using beat-synchronous features.

3.5.2.3 Influence of the Position of an Adaptive Window

We now consider the case of a *beat-adaptive analysis*. The choice of the position of the window according to the beat location is not trivial. In our view, the most logical solution would be to center the analysis window exactly between two beats. In this case, the problem of mixing several harmonies within the same beat does not exist. However, experiments have shown that the best position of the adaptive window depends on the music style (see Section 3.6.3.2).

3.6 Selecting a Feature Vector for Harmonic Analysis

In this Section, we evaluate and compare the three presented chroma representations. We do not intend to compare the numerous chroma feature extraction methods proposed in the literature but we intend to draw some general conclusions concerning the use of chroma features extracted either from a fixed resolution analysis, a multi-resolution analysis or a multi-f0 pitch tracking approach.

3.6.1 Defining a Measure to Compare Various Features

Selecting an input feature among others is a complex task. First of all, we need to define the criteria that are relevant for comparison. The potential superiority of a feature above another depends on their final use. The various presented features exhibit different weaknesses and strengths. Choosing one input feature among the others is a result of a compromise. We aim here at selecting the most appropriate chroma representation for the harmony related tasks (key or chord recognition). The best candidate should provide the most reliable information about the notes that comprise the played chord.

The problem of selecting the best front-end representation for a given task has already been studied in some previous works. For instance [Dav07] compares the performances of a downbeat tracker using three different spectral representations (a constant-Q spectrogram, a 36-bin chromagram and a 12-bin chromagram), [MEK09] compares the robustness to timbre changes of a newly proposed chroma representation with some commonly used chroma types including two freely available chroma representations [3].

However, we are not aware of any systematic analysis and comparison of the large number of previously proposed chroma representations except two recent studies that investigate the use of various chroma representations. [VPM08] investigate six formerly proposed algorithms and proposes a new scheme based on multipitch tracking for chroma vector computation. [SSG+09] analyzes and compares different methods for audio chroma feature extraction. These two studies lead to different results concerning the performances of the chroma features. For instance, in [SSG+09], the Enhanced Pitch Class Profiles originally proposed in [Lee06a] are found to perform better than the Instantaneous Frequency spectrum-based chroma vectors [EP07], whereas the opposite conclusion is claimed in [VPM08]. This can be explained by the fact that different performance measures and different evaluation test-sets are used in the two studies. This shows that comparison between chroma features is not a trivial point.

We investigate here the three above-mentioned chroma-based representations (FFT, CQT, $f0$) through the analysis of experimental results obtained on a number of music audio excerpts.

[3] Ellis: "Chroma features analysis and synthesis," http://www.ee.columbia.edu/dpwe/

3.6.1.1 Previously proposed measures

In the literature, we can distinguish between two approaches for comparing front-end features. On the one hand, the features are compared through the results of an application. For instance, [DBSD04] compares the effects of using a fixed resolution spectral analysis or a multi-resolution subband approach in the context of onset detection. To evaluate and compare the two methods, a measure of onset detection accuracy is defined and computed over a set of recordings. [NM06] proposes a study that investigates the effects of low-level digital signal processing parameters (such as the analysis window length) for an HMM-based key estimation algorithm. One set of parameters is selected as a reference setting. The effect of changing the other parameter values is measured by evaluating key estimation performance of the algorithm on two different test-sets of real audio recordings (110 Beatles songs plus 48 piano recordings of all preludes and fugues from J.S. Bach). [Dav07] investigates the use of three beat-synchronous spectral representations for detecting bar boundaries based on harmonic changes ((i) a constant-Q spectrogram (ii), a 36 bin-chromagram; and (iii) a 12-bin chromagram). The comparison between the different spectral front-ends is done through the downbeat tracking performance results.

The drawback of these approaches is that the features are compared through a complex process in which, in general, not only the type of features used as front-end are evaluated but also other parameters that have an impact on the result. It is thus difficult to analyze the results and to distinguish between differences due to the type of the feature and differences due to other parameters.

On the other hand, features can be compared using a specific evaluation measure that is supposed to measure their quality. In [VPM08], a large-scale experimental evaluation is performed to compare a newly proposed chroma representation based on multiple pitch tracking techniques with six other schemes. The experimental evaluation is performed by measuring the similarity of the novel and the previous chroma representation with "ideal" profiles retrieved from manually labeled chords on a data set consisting of 161 30s-length real audio excerpts covering different tempi and genres. The goal of the experimentation is to quantify the closeness between each computed chroma profile and the annotated chord profile. It relies on the argument that the better the resemblance is, the more accurate the chord detection will be. To quantify the similarity between the computed chroma profiles and the annotated chord profiles, for each chord segment, the computed chroma V_c is compared to the annotated chroma vector V_a consisting of 1 when the note belongs to the chord and 0 otherwise using a cosine similarity distance S defined as

$$S(V_c, V_a) = \frac{(V_c|V_a)}{\|V_c\|\|V_a\|}$$

To measure the quality of the tested algorithm two measures are used: the mean cosine similarity across all chord segments and the mean reciprocal rank (MRR) [EP07]. The various chroma representations are ranked according to their mean cosine similarity.

[SSG+09] analyzes and compares different methods for audio chroma feature extraction using 55 audio tracks synthesized from MIDI files. This database is built considering four parameters: pitch, chord type, duration and attack. Evaluation of the chroma feature

extraction methods is done through a set of measures that are related to the so-called *Chroma Precision* (CP). For each computed feature vector V_c, the intensities of the pitch classes corresponding to the tonal content of the input signal are added:

$$CP(V_c) = \frac{\sum_{i=1}^{12} I(V_c) * s(i))}{\sum_{i=1}^{12} I(V_a)}, s(i) \in [0, 1]$$

This evaluation measure is close to the one proposed in [VPM08] since the chroma is considered all the more precise when it is close to a bit mask representing the tonal content of the input signal.

We do not agree with the argument that "the better the resemblance is, the more chance there is that the computed chroma profile can give rise to an accurate chord detection and classification" [VPM08]. Indeed, the goal of chord estimation is to select a chord among a set of chord candidates. Resemblance with the annotated chord is thus only important regarding resemblance with the other possible chords. A greater resemblance with an annotated chord does not result automatically in an improved accuracy of chord detection (see an example in Section 3.6.1.2, Figure 3.13. In other words, the discriminative power of the chroma vectors must be taken into account in the evaluation measure.

To measure the quality of the various representations regarding chord estimation, we should consider the following three conditions:

1. (i) First, the notes present in the chord should be clearly emphasized in the chroma feature.

2. (ii) Secondly, the similarity between the computed chroma feature and the chord templates that do not correspond to the annotated (ground-truth) chord should be weak.

3. (iii) Finally, the similarity between the computed chroma feature and all possible chord templates should be maximum with the template corresponding to the annotated chord.

The best chroma feature should be thus selected as the one that gives the maximum discriminative power.

The idea of measuring the performance of a feature extraction method in relation to its discriminative power is presented in [MEK09]. In this paper, a method for making chroma features more robust to changes in timbre and instrumentation is presented. The novel chroma feature is quantitatively compared with three commonly used chroma types that serve as reference. Two types of experiments are conducted.

1. The first experiment is conducted on synthesized audio. A MIDI file containing various chords is synthesized into 24 different ways using 8 different instruments playing the file in 3 different octaves and considering two cases: the attack and the sustain phase. A class composed of the 48 computed chroma vectors is formed for each chord. The distance between two chroma vectors is computed using the cosine

distance. Three measures are computed to quantify the degree of timbre invariance of a given chroma type: the within-class distance μ_1 (corresponding to the average over the distances computed between any two chroma vectors that belong to the same class) that measures the degree of timbre invariance and the between-class distance μ_0 (corresponding to the average over the distances computed between any two chroma vectors from different chord chroma classes) that measures the discriminative power of the chroma representation. Finally, the inertia ratio $\rho = \frac{\mu_1}{\mu_0}$ expresses the across-class distance relative to the within-class distance.

2. The second set of experiments is conducted on real audio data. The newly proposed chroma features are compared to the previously proposed ones by means of several performance measures that allow comparing a query sequence with a given database sequence.

3.6.1.2 Proposed Measure for Chroma Feature Comparison

To compare the different chroma feature extraction methods, we propose a measure that quantifies the quality of a chroma vector in terms of representation of the harmonic content of the signal.

From the previous observations, in order to measure the quality of the various proposed features, we follow the approaches proposed in [SSG+09] and [VPM08] and compare each computed chroma feature with the input signal using a bit mask composed of zeros and ones that represents the tonal content of the input signal (the ground-truth chord). The chord template contains a 1 if the pitch class belongs to the chord and a 0 if it does not. For instance, a C major chord template (C-E-G) has the following format: [1,0,0,0,1,0,0,1,0,0,0,0][4].

We use a measure inspired from the one proposed in [MEK09] to quantify the resemblance between the computed and the theoretical chroma against the resemblance between the computed chroma vector and the other possible chords. In our experiments, we consider only the 24 major and minor triads. The chord templates are denoted by $T_i, i \in [1:24]$. Distances between the computed chroma vectors and the theoretical chord templates are computed using a cosine similarity distance, as in [VPM08]. We restrict here our analysis to this commonly used distance measure but it is important to note that the type of the distance used to compare two chroma features has an impact on the results, as shown in [OGF09a]. We plan to pay more attention to this point in future works.

Let us consider a given input audio chord corresponding to an "ideal" template T_i and let C denote a chroma vector computed on this audio signal. According to condition (i) the chroma feature should match as closely as possible the theoretical template corresponding to the chord. The correct-chord distance D_{CC} is computed as:

$$D_{CC}(C) = \frac{C.T_i}{\|C\|\|T_i\|} \tag{3.23}$$

[4]Note that for the sake of simplicity, we do not consider here the problem of harmonics evoked in paragraph 3.2.2.1 and we do not consider harmonics in the theoretical templates that represent the input signal. We will give more attention to this issue in the next chapter.

We now consider condition (ii). We aim to find a measure that characterizes the discriminative power of a chroma representation. Our first idea was to use a measure similar to the between-class distance employed in [MEK09]: the average distance over any computed chroma vector C and any chord template that does not correspond to the annotated chord:

$$D_{av}(C) = \frac{1}{23}\sum_{j\neq i}\frac{C.T_j}{\|C\|\|T_j\|} \tag{3.24}$$

However, this measure does not reflect the discriminative power of a chroma representation. Indeed, the computed chroma vector might be even more similar to an other chord than to the annotated chord although the value D_{av} is small.

To illustrate this, consider Figure 3.13, which represents a multi-f0 based chroma representation (top, left) and a CQT based chroma representation (bottom, left) of an F major chord (F-A-C) extracted from the Beatles song *Misery*. Let us denote these two vectors by C_{f0} and C_{CQT} respectively. The right part of the figure represents the values of the correct-chord distance computed between the extracted chroma and the 24 chord templates.

It can be seen that the value $D_{CC}(C_{f0}) = 0.9120$ is much larger than the value $D_{CC}(C_{CQT}) = 0.6382$. However, the amplitude of the A note in C_{f0} is very small (this is probably due to the fact that the considered frame is disrupted by a drum sound that makes multi-f0 estimation difficult). As a result, the computed chroma vector is closer to an Fm chord than to a FM chord (see the dashed circle in the right part of Figure 3.13, top). On the contrary, the FM chord is well discriminated from the others in the case of C_{CQT}.

Let us compute the distance D_{av} for the two representations. For C_{f0}, we obtain a value of 0.2212 and a ratio $\frac{D_{CC}}{D_{av}} = 4.1227$. This is much larger than the value $\frac{D_{CC}}{D_{av}} = 2.7378$ obtained in the case of C_{CQT}, although the annotated chord is clearly better discriminated using the constant-Q based approach. The poor discriminative power of C_{f0} over C_{CQT} in the example is not represented using the average distance D_{av} between the computed chroma vector and any chord template that does not correspond to the annotated chord.

To take into account the discriminative power of the chroma features (condition (ii)), we define the incorrect-chord distance D_{IC} as:

$$D_{IC}(C) = \max_{j\neq i}\frac{C.T_j}{\|C\|\|T_j\|} \tag{3.25}$$

The ratio $D_{CIC} = \frac{D_{CC}}{D_{IC}}$ expresses the correct-chord distance relative to the incorrect-chord distance. The mean value of the distances D_{CIC} computed over all the frames of the test database, denoted by $\bar{D_{CIC}}$ is used to measure the quality of a given chroma representation. A good chroma representation should result in a large value of $\bar{D_{CIC}}$.

Finally, to take condition (iii) into account, we also compute the rate of correctly detected chords using a given chroma representation, which is given by the percentage of chords for which the similarity between the computed chroma feature and the chord templates is maximum for the template corresponding to the annotated chord. Note that this is equivalent to the condition $D_{CIC} > 1$. In the following tables of results, this will

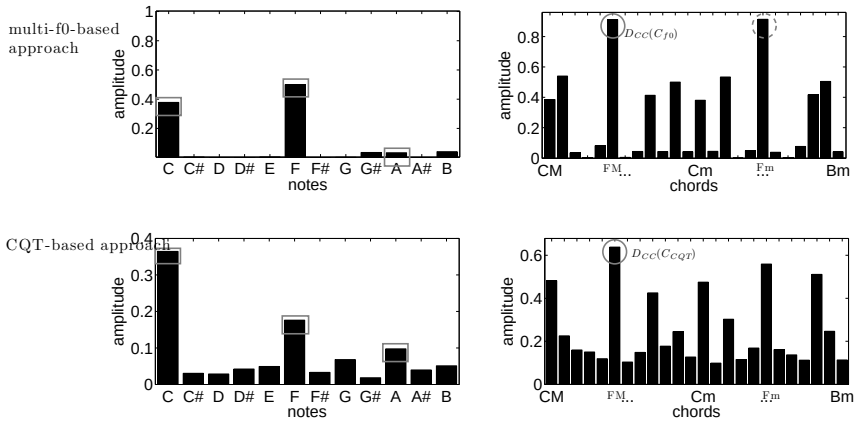

Figure 3.13: Chroma representations of a AM chord and similarity with the chord templates using a multi-f0 based approach [top] and using a CQT based approach [bottom]. The grey rectangles indicate the amplitude of the notes composing the FM chord. The grey circles indicate the D_{CC} values.

be referred to as "% of correct chords".

3.6.2 Database for Feature Selection

The various presented chroma feature extraction methods were analyzed and compared through a database consisting of a number of short excerpts of about 20 seconds extracted from audio recordings and hand-labeled in chords (major and minor triads) by the author. The chords are annotated against a time grid defined by the beats. The database is divided into non-percussive audio (*DATClas*) and percussive audio (*DATPop*). *DATClas* corresponds to extracts of classical music with various instruments and *DATPop* corresponds to audio excerpts of popular and rock music containing voices and drum sounds. These two databases are described in details in Chapter 2, Section 2.3.1.

3.6.3 On the use of a Beat-Synchronous Analysis

In this section, we give a quantitative analysis of the use of beat-synchronous chroma features. We first discuss the effect of using beat-synchronous chroma features instead of frame-by-frame features in terms of capturing the harmonic content of a piece. We then study the influence of the position of an adaptive window according to the beat positions.

3.6.3.1 Beat-Synchronous Versus Frame-by-Frame Analysis

The results of comparison between frame-by-frame versus beat-synchronous analysis for the various considered methods are given in Table 3.4. We give the mean value and the standard deviation of the various evaluation measures computed over all the frames of the test-set. The chord estimation results are illustrated in Figure 3.14.

Table 3.4: Similarity measure for comparing frame-by-frame versus beat-synchronous (BS) analysis. *DATPop*: Popular music database, *DATClas*: Classical music database. SS: Statistical significance.

	DATPop							
	$f0$	$f0B_{AV}$	FFT_L	FFT_LB_{AV}	FFT_S	FFT_SB_{AV}	CQT	$CQTB_{AV}$
$\bar{D_{CC}}$	0.6361 ± 0.0971	0.6347 ± 0.0968	0.4921 ± 0.0751	0.4928 ± 0.0757	0.4423 ± 0.0500	0.4416 ± 0.0501	0.5118 ± 0.0683	0.5133 ± 0.0679
$\bar{D_{IC}}$	0.8209 ± 0.0375	0.6503 ± 0.0776	0.4872 ± 0.0666	0.4768 ± 0.0634	0.4812 ± 0.0429	0.4622 ± 0.0384	0.5120 ± 0.0618	0.4922 ± 0.0550
D_{CIC}	0.7835 ± 0.0946	0.9723 ± 0.0767	1.0078 ± 0.0520	1.0297 ± 0.0493	0.9269 ± 0.0608	0.9590 ± 0.0609	0.9973 ± 0.0540	1.0399 ± 0.0455
% Correct	26.3954± 8.6968	59.6364± 20.1500	63.5549± 11.2007	66.8039± 12.8174	46.9668± 14.4024	49.8616± 15.2406	61.5483± 11.8779	70.3990± 10.2607
SS	yes		yes		yes		yes	

	DATClas							
	$f0$	$f0B_{AV}$	FFT_L	FFT_LB_{AV}	FFT_S	FFT_SB_{AV}	CQT	$CQTB_{AV}$
$\bar{D_{CC}}$	0.7349 ± 0.0584	0.7358 ± 0.0577	0.6030 ± 0.0893	0.6046 ± 0.0856	0.5857 ± 0.0755	0.5844 ± 0.0717	0.6159 ± 0.0768	0.6185 ± 0.0738
$\bar{D_{IC}}$	0.8075 ± 0.0438	0.7107 ± 0.0343	0.6037 ± 0.0516	0.5881 ± 0.0504	0.5969 ± 0.0387	0.5750 ± 0.0346	0.6162 ± 0.0404	0.5955 ± 0.0431
D_{CIC}	0.9280 ± 0.1129	1.0435 ± 0.1220	1.0004 ± 0.0741	1.0300 ± 0.0771	0.9884 ± 0.0813	1.0199 ± 0.0862	1.0045 ± 0.0703	1.0412 ± 0.0722
% Correct	39.4648± 21.5594	73.4211± 15.1892	66.4417± 12.7790	70.7845± 13.0598	64.6579± 15.0906	68.5439± 16.4696	67.6692± 12.6419	73.1704± 14.7149
SS	yes		yes		yes		yes	

It can be seen that, for all the methods, the use of beat-synchronous features improves the results. Using a paired sample t-test, we found the difference between the results of the beat-synchronous and frame-by-frame analysis to be statistically significant at the 5% level[5].

This corroborates the results obtained by Bello & Pickens in [BP05]: the use of beat-synchronous analysis frames helps overcome noise introduced by transient components in the sound, short ornamentations and passing notes. Averaging the analysis windows between two beats results in some smoothing. Of course, we need for this that the beat positions are correctly detected. This may not be the case in real situations. In [Bel07], Bello compares beat-synchronous with frame-based chroma features for the purpose of cover song retrieval. It is found that, due to errors in the beat

[5]The number of analysis frames is different between the beat-synchronous and the frame-by-frame analysis. We thus computed a score for each audio excerpt and performed the t-test using each excerpt as a sample.

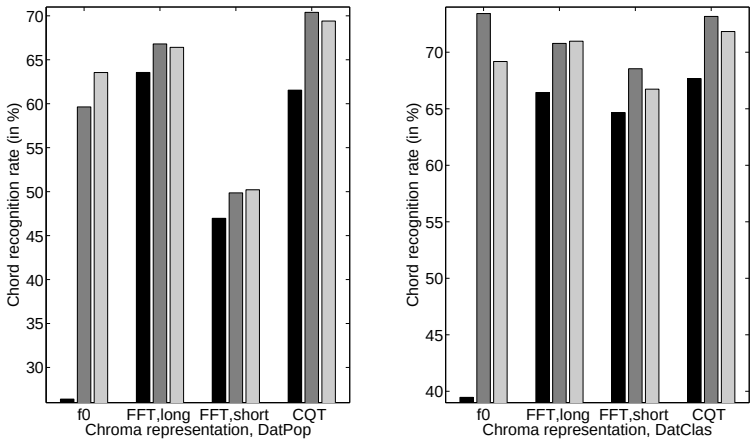

Figure 3.14: Chord estimation results on *DATPop* (left) and *DATClas* (right) for frame-by-frame (black bars) versus beat-synchronous (B_{AV}) analysis: B_{AVmean} in dark grey, $B_{AVmedian}$ in light grey.

tracking, the frame-based analysis consistently outperforms the beat-synchronous analysis.

It can be seen that the use of long analysis windows (cases FFT_L and CQT for low frequencies) leads to higher chord estimation results than when using a short analysis window, even if more undesirable information from the adjacent chords is captured by the beat-synchronous chroma vectors. This is probably due to the fact that we need sufficiently long windows to capture the bass notes, which are in general the most important information for chord estimation.

Figure 3.14 shows that the results are different according to whether the beat-synchronous features are computed with method B_{AVmean} (mean) or with method $B_{AVmedian}$ (median). However, the results are not statistically significant. Tests on a larger database would be required to possibly decide which method is the best.

3.6.3.2 Influence of the Position of an Adaptive Window

In this section, we present some experiments that we conducted to study the influence of the position of the beat-adaptive window according to the beat positions. The different tested center positions are represented in Figure 3.15.

Figure 3.16 and Table 3.5 present the results obtained when centering the beat synchronous window on different positions, using the proposed evaluation measures. Note that the adaptive analysis window corresponding to a pair of successive beat positions b_k and b_{k+1} has a length of $b_{k+1} - b_k$.

In the case of popular music (*DATPop*), the best results are obtained when the

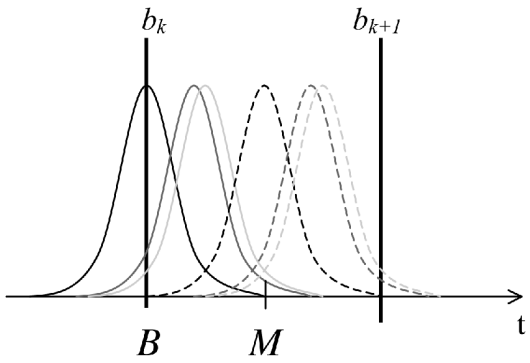

Figure 3.15: Different tested center positions of the beat-adaptive window: on the beat (B), on the beat plus a $\frac{1}{5}$-beat duration shift, on the beat plus a $\frac{1}{4}$-beat duration shift (solid lines), on the middle of the two beats (M), on the middle of the two beats plus a $\frac{1}{5}$-beat duration shift, on the middle of the two beats plus a $\frac{1}{4}$-beat duration shift (dashed lines).

Table 3.5: Similarity measure between extracted chroma features and chord templates and percentage of chords that have been correctly detected for investigating the influence of the position of beat adaptive windows. B: on the beat, M: between two beats (middle). *DATPop* : Popular music database, *DATClas*: Classical music database.

	DATPop					
	B	B + $\frac{1}{5}$	B + $\frac{1}{4}$	M	M + $\frac{1}{5}$	M+ $\frac{1}{4}$
$\bar{D_{CC}}$	0.4471 ± 0.0759	0.4815 ± 0.0848	0.4946 ± 0.0821	0.5089 ± 0.0672	0.4977 ± 0.0713	0.4911 ± 0.0739
$\bar{D_{IC}}$	0.4603 ± 0.0687	0.4846 ± 0.0715	0.4964 ± 0.0702	0.5107 ± 0.0604	0.5204 ± 0.0667	0.5183 ± 0.0684
$\bar{D_{CIC}}$	0.9710 ± 0.0507	0.9906 ± 0.0596	0.9942 ± 0.0600	0.9952 ± 0.0454	0.9535 ± 0.0650	0.9457 ± 0.0718
% Correct	53.6962 ± 11.1792	58.1196 ± 11.8082	60.1012 ± 12.6013	61.8671 ± 10.5671	53.4751 ± 8.4631	52.4971 ± 8.5302

	DATClas					
	B	B + $\frac{1}{5}$	B + $\frac{1}{4}$	M	M + $\frac{1}{5}$	M+ $\frac{1}{4}$
$\bar{D_{CC}}$	0.5478 ± 0.0398	0.6133 ± 0.0482	0.6244 ± 0.0561	0.6219 ± 0.0893	0.6272 ± 0.0910	0.6208 ± 0.0909
$\bar{D_{IC}}$	0.5906 ± 0.0463	0.6221 ± 0.0282	0.6255 ± 0.0287	0.6283 ± 0.0400	0.6304 ± 0.0418	0.6201 ± 0.0443
$\bar{D_{CIC}}$	0.9413 ± 0.0587	0.9918 ± 0.0652	1.0058 ± 0.0809	0.9939 ± 0.0950	1.0027 ± 0.0843	1.0083 ± 0.0796
% Correct	56.2206 ± 6.5431	66.1654 ± 11.7777	70.4536 ± 14.6333	68.3559 ± 17.5472	67.2632 ± 13.3461	67.4561 ± 13.4273

window is centered exactly between two beats. Using a paired sample t-test, we found the difference between window positions B and M to be statistically significant at the 5% level. Moreover, the more information from adjacent chords is taken into account, the worst the results are. The huge increase in the results from position B to position $B + \frac{1}{5}$ may be due to the fact that, by placing the center of the window not exactly on the beat,

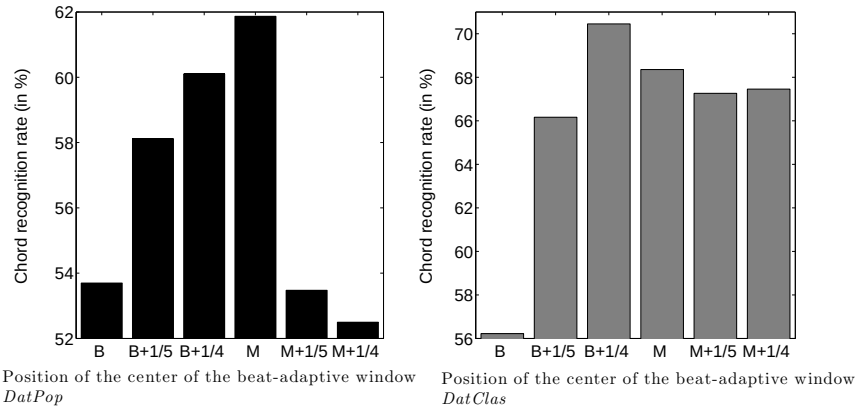

Figure 3.16: Results of comparison of an adaptive window analysis. On the beat (B), on the beat plus a $\frac{1}{5}$-beat duration shift, on the beat plus a $\frac{1}{4}$-beat duration shift, on the middle of the two beats (M), on the middle of the two beats plus a $\frac{1}{5}$-beat duration shift, on the middle of the two beats plus a $\frac{1}{4}$-beat duration shift. Black bars (left) correspond to the results obtained on *DATPop* and grey bars (right) correspond to the results obtained on *DATClas*.

we avoid taking into account part of the noise introduced by transient components in the sound.

It is interesting to notice that the results are different in the case of classical music (*DATClas*). In this case, the best results are not obtained when the window is centered exactly between two beats. A deeper analysis shows that the best position of the adaptive window depends on the music style.

In our test-set, we have 6 excerpts of piano Mozart sonatas. Each beat can be associated with a chord. Notes composing the chord are played in general on the beat, and ornamental notes, passing notes or scales that do not belong to the chord usually follow them. As a result, the chroma features computed between two beat positions capture some harmonic information that does not correspond to the underlying harmony.

For all of the other classical music pieces, the best results are obtained when the window is centered exactly between two beats (case M). We can also notice that the worst results are obtained when the center window is positioned on the beats B. This corresponds to the case where the most information from adjacent chords is taken into account.

3.6.3.3 Conclusion on Beat-Synchronous Analysis

Many algorithms related to music content analysis rely on beat-synchronous features. We have investigated the consequences of using beat-synchronous chroma features for harmonic content analysis. We have shown that it increases the chord estimation results

under the assumption of perfect beat tracking. Analysis and experiments show that it is necessary to make a trade-off between having a satisfying frequency resolution and mixing the harmonic content of two different chords in one beat-synchronous chroma vector. We have also shown that in the case of a beat-adaptive analysis, the choice of the window position depends on the music style.

3.6.4 Fixed versus Multi-resolution Analysis

The results of comparison between fixed versus multi-resolution chroma feature extraction over databases *DATClas DATPop* are represented in Table 3.6, in which we give the mean value and the standard deviation of the various distances computed over all the beat-synchronous frames of the test-set. They are illustrated in Figure 3.17.

Table 3.6: Similarity measure between extracted chroma features and chord templates and percentage of chords that have been correctly detected for comparing fixed versus multi-resolution chroma feature extraction. From left to right: FFT_LB_{AV} FFT using a long analysis window (0.5s), FFT_SB_{AV} FFT using a short analysis window (0.125s), $FFTB_{AD}$ beat-synchronous FFT, $CQTB_{AV}$ CQT. SS: Statistical significance between the considered FFT-based approach and the CQT-based approach.

	DATPop			
	FFT_LB_{AV}	FFT_SB_{AV}	$FFTB_{AD}$	$CQTB_{AV}$
$\bar{D_{CC}}$	0.4928 ± 0.0757	0.4416 ± 0.0501	0.5089 ± 0.0672	0.5133 ± 0.0679
$\bar{D_{IC}}$	0.4768 ± 0.0634	0.4622 ± 0.0384	0.5107 ± 0.0604	0.4922 ± 0.0550
$\bar{D_{CIC}}$	1.0297 ± 0.0493	0.9590 ± 0.0609	0.9952 ± 0.0454	1.0399 ± 0.0455
SS	yes	yes	yes	
% Correct	66.8039 ± 12.8174	49.8616 ± 15.2406	61.8671 ± 10.5671	70.3990 ± 10.2607

	DATClas			
	FFT_LB_{AV}	FFT_SB_{AV}	$FFTB_{AD}$	$CQTB_{AV}$
$\bar{D_{CC}}$	0.6046 ± 0.0856	0.5844 ± 0.0717	0.6219 ± 0.0893	0.6185 ± 0.0738
$\bar{D_{IC}}$	0.5881 ± 0.0504	0.5750 ± 0.0346	0.6283 ± 0.0400	0.5955 ± 0.0431
$\bar{D_{CIC}}$	1.0300 ± 0.0771	1.0199 ± 0.0862	0.9939 ± 0.0950	1.0412 ± 0.0722
SS	yes	yes	yes	
% Correct	70.7845 ± 13.0598	68.5439 ± 16.4696	68.3559 ± 17.5472	73.1704 ± 14.7149

As explained above, a fixed-resolution analysis is the result of a trade-off between a good temporal resolution (short analysis window length) and a good spectral resolution (long analysis window length). The results presented in Table 3.6 show that the CQT-based approach outperforms the FFT-based approach, especially in the case of percussive music.

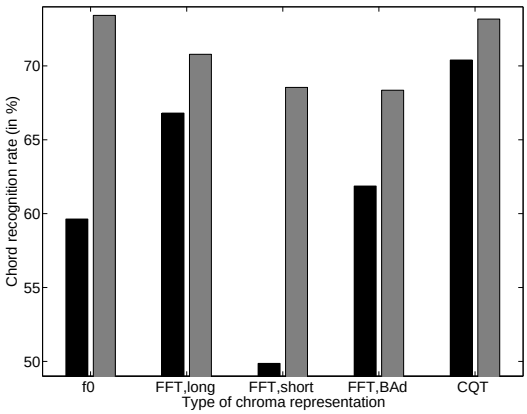

Figure 3.17: Results of comparison between various chroma-based representations. Black: *DatPop*, grey: *DatClas*.

We performed a paired sample t-test to determine whether there is a significant difference between the results obtained with the two approaches. The null hypothesis could be rejected at the 5% significance level, which indicates that the FFT based features are outperformed by the CQT based features. The differences, although small, are statistically significant.

We illustrate these results on an example. Let us consider the beginning of the Beatles song *Love Me Do*. Figure 3.19 and 3.18 represent respectively the chromagram and the semitone pitch spectrum of the first seconds of this song. The harmony is waving between C major and G major chords. In the case of the CQT, all chord changes are correctly detected whereas in the case of the FFT, the C major chords are considered as G major chords.

If we listen to the music, we can hear that the harmony given by the accompaniment is covered by the melody played by the harmonica. When C major (C-E-G) chords occur, the bass (the C note) is hardly audible. The duration of the C2 midi note played by the bass is very short.

- **FFT long analysis window:** We consider a chroma feature extraction based on a FFT using a long analysis window length of 0.5s. We can see in the left part of Figure 3.19 that the C note of the first C major chord is not accurately discriminated from the other pitch classes on the chromagram. Looking at the semitone pitch spectrum (see left part of Figure 3.18), it can be seen that the semitone pitch-class spectrum is blurred (due to the percussive sounds).
- **FFT short analysis window:** We consider the case where a smaller analysis window is used. It is now set to 125ms. The semitone pitch-class spectrum and the chromagram are respectively represented in the middle part of Figures 3.18 and 3.19. It can be seen that the C2 note is not detected anymore (see the 21^{st} frame of the chromagram). This is because the frequency resolution is too low.

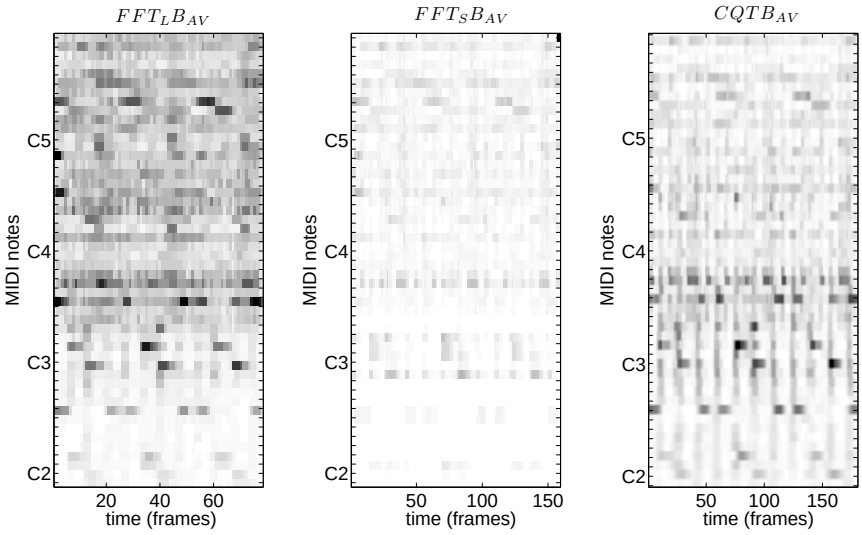

Figure 3.18: Pitch class spectrum of the first seconds of the song *Love Me Do*. From left to right: FFT_LB_{AV}, FFT_SB_{AV} and $CQTB_{AV}$.

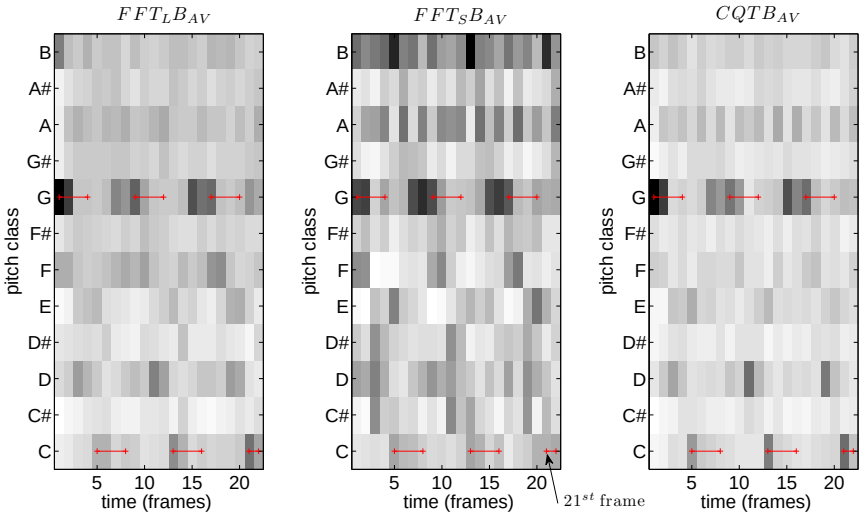

Figure 3.19: Chromagram of the first seconds of the song *Love Me Do*. From left to right: FFT_LB_{AV}, FFT_SB_{AV} and $CQTB_{AV}$. The horizontal lines correspond to the annotated chords (ground truth).

- **Constant-Q transform:** The use of a constant-Q transform to compute the chromagram allows a better management of the time-frequency trade-off problem. The use of long windows in low frequency allows detecting accurately the bass line (G-D-C) whereas the use of short windows in higher frequencies allows reducing the effects of percussive sounds. This is illustrated in the right part of Figure 3.19.

This example illustrates that the multi-resolution based approach can be an answer

to the trade-off between the temporal and spectral resolution of the FFT. The constant-Q based approach itself presents some drawbacks (for instance regarding the problem of computing beat-adaptive related features, as explained below) but it seems a more powerful chroma feature, especially for popular and rock music, that contains in general lots of percussive sounds.

3.6.5 Multi-f0s Versus Spectral Representation

Table 3.7 presents the results obtained using a multi-f0s based approach for chromagram computation and using a constant-Q based approach. See also Figure 3.17.

Table 3.7: Similarity measure between extracted chroma features and chord templates and percentage of chords that have been correctly detected for comparing multi-f0s (left) versus constant-Q based chroma feature extraction.

	DATPop		
	multi-f0s	CQT	Statistical significance
$\bar{D_{CC}}$	0.6347 ± 0.0968	0.5133 ± 0.0679	
$\bar{D_{IC}}$	0.6503 ± 0.0776	0.4922 ± 0.0550	
$\bar{D_{CIC}}$	0.9723 ± 0.0767	1.0399 ± 0.0455	
% Correct	59.6364 ± 20.1500	70.3990 ± 10.2607	yes

	DATClas		
	multi-f0s	*CQT*	Statistical significance
$\bar{D_{CC}}$	0.7358 ± 0.0577	0.6185 ± 0.0738	
$\bar{D_{IC}}$	0.7107 ± 0.0343	0.5955 ± 0.0431	
$\bar{D_{CIC}}$	1.0435 ± 0.1220	1.0412 ± 0.0722	
% Correct	73.4211 ± 15.1892	73.1704 ± 14.7149	no

It is difficult to decide which one of the chroma representations based on multi-f0s and constant-Q transform is the best. Both have proved in previous works to give good results as shown during the MIREX 2008 audio chord detection contest where a method using a CQT-based chroma representation [BP05] and a method using a f0-based chroma representation [RK08a] were among the three approaches that gave the best results. Note that the multi-f0 based approach is more recent (probably because it follow the advances of multi-f0 estimation) and has been used in a smaller number of works than the CQT-based approach. It can be seen in Table 3.7 than in the case of non-percussive audio, the two representations yield close results. We performed a paired sample t-test to determine whether

there is a significant difference between the results obtained with the two approaches. In the case of classical music, the null hypothesis could not be rejected at the 5% significance level, which indicates that the multi-f0 based features are not outperformed by the CQT-based features in the case of non-percussive music.

However, in the case of percussive audio (popular and rock music), the constant-Q-based chroma features clearly outperform the multi-f0 based chroma features. A deeper analysis of the results shows that multi-f0-based chroma features computed on the pieces containing a lot of drum sounds give particularly low results as compared to the CQT. This is because the estimation of the multi-f0 is less accurate in case of percussive audio containing transient and noise.

Regarding the results obtained in the case of non-percussive audio, we believe that the multi-f0 approach is very promising. However, to be usable in the case of percussive audio, the signal should be pre-processed before computation to reduce transients and noise. A separation between the harmonic and drum parts would probably lead to a successful use of multi-f0 based chroma features. This has been corroborated by some preliminary experiments that we conducted on the popular music test database. We have intended to reduce the transients in the signal using the IRCAM software AudioSculpt [6]. The results (for the f0-based and the CQT-based chroma features) are presented in Table 3.8 and illustrated in Figure 3.20. It can be seen that the performances of the chroma features seem to be improved using this transient reduction pre-processing step. However, the results are not statistically significant for the multi-f0 based method. The problem of reducing transients and noise deserves a full attention and this is left for future works.

Table 3.8: Similarity measure between extracted chroma features and chord templates on the popular music database for comparing CQT and $multi-f0$-based chroma features using (TR) or not a transient reduction pre-processing step. SS indicates statistical significance between the two cases. We also indicate the percentage of chords that have been correctly detected.

	CQT	CQT_{TR}	SS_{CQT}	$f0$	$f0_{TR}$	SS_{f0}
$\bar{D_{CC}}$	0.5133 ± 0.0679	0.5634 ± 0.0600		0.6347 ± 0.0968	0.6465 ± 0.0804	
$\bar{D_{IC}}$	0.4922 ± 0.0550	0.5412 ± 0.0512		0.6503 ± 0.0776	0.6567 ± 0.0713	
$\bar{D_{CIC}}$	1.0399 ± 0.0455	1.0413 ± 0.0448		0.9723 ± 0.0767	0.9864 ± 0.0554	
% Correct	70.3990 ± 10.2607	74.3661 ± 8.2053	yes	59.6364 ± 20.1500	62.9956 ± 17.5220	no

We have seen that in the case of percussive audio, the constant-Q based chroma features clearly outperform the multi-f0 based chroma features. In the rest of this PhD thesis, we will thus work using constant-Q based chroma features. Another argument that motivated or choice is that the multi-f0 estimation of a music track and thus the chroma features based on the f0s is a very costly process in terms of computation time

[6] AudioSculpt is an application for the musical analysis and processing of sound files developed at IRCAM since 1993.

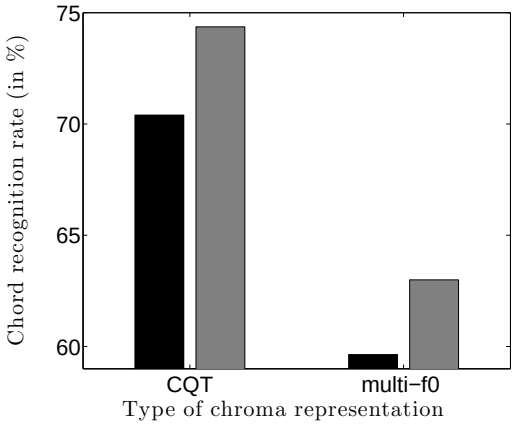

Figure 3.20: Performances in terms of correct chord recognition rate (in %) of chroma features using a transient reduction pre-processing step in the case of a CQT-based analysis (left) and an multi-f0-based analysis (right), for the popular music database. Black bars correspond to the results obtained without a pre-processing step and grey bars correspond to the results obtained in the case of a transient reduction pre-processing step.

compared to the computation of chroma features based on the CQT.

3.7 Summary and Conclusion

At the front-end of our models, we extract a *chromagram*, a representation of the signal that captures its harmonic content. We explored several schemes for chromagram computation and investigated several issues related to the use of each representation (problem of noise, beat-synchronous features). We conducted a number of experiments on short audio excerpts and proposed some evaluation measures that allow the comparison between the various representations.

We have shown that the use of a beat-synchronous analysis increases the chord estimation results under the assumption of perfect beat tracking. Analysis and experiments show that it is necessary to make a trade-off between having a satisfying frequency resolution and mixing the harmonic content of two different chords in one beat-synchronous chroma vector. We have also shown that in the case of a beat-adaptive analysis, the choice of the window position depends on the music style.

The Constant-Q based chroma features were preferred to the FFT based chroma features. They were found to reflect more accurately the harmonic content than the FFT-based chroma features, especially for popular and rock music that contain lots of percussive sounds: the use of long windows in low frequency allows detecting accurately the bass line, which is very important for chord estimation, whereas the use of short windows in higher frequencies allows reducing the effects of percussive sounds.

Tests on classical piano music showed that the use of multi-f0 features seems to be a

promising approach for harmonic content description. However, we did not find this representation convenient for our system since we do not currently have any harmonic/noise separation front-end and thus percussive sounds and noise disrupt the multi-f0s estimation, especially in popular music. Moreover, the rest of our system is computationally very efficient compared to the multi-f0 analysis (far less time-consuming). We thus did not favor the use of multi-f0 based chroma features in the rest of our work.

Chapter 4

Chord Progression Estimation From an Audio File

In this chapter, we focus on the problem of the automatic estimation of the chord progression from an audio file using chroma features as observation of the music signal. From the audio signal, a set of chroma vectors representing the pitch content of the file over time is extracted. The chord progression is then estimated from these observations using hidden Markov models. Several methods are proposed that allow taking into account music theory, perception of key and presence of higher harmonics of pitch notes. They are evaluated and compared with existing algorithms through a large-scale evaluation on 110 hand-labeled songs from the Beatles.

Contents

4.1 Introduction

This chapter is devoted to chord progression estimation. Chords are central to our work. In the global model for musical attribute estimation presented in this dissertation, we consider harmony as the core around which other musical attributes are organized.

In this chapter, we review and analyze several previous methods for estimating the chord progression of a piece of music directly from audio signals of musical recordings. The presented methods are based on chroma features and hidden Markov models (HMMs). We then propose improvements of these methods to build our chord estimation algorithm that will serve as a basis for investigating interaction between various musical attributes in the next chapters. The presented work is based on the publication [PP07]. The major contributions of this chapter are:

1. We provide a detailed review of the previous works in the area of chord estimation.
2. We compare and extend some previous proposed methods for chord estimation.
3. We propose a new method to take into account the problem of harmonics in the case of chord estimation.
4. We compare several previously used state transition matrices with newly proposed ones in the HMM.
5. We present a large-scale evaluation of the proposed chord estimation systems.
6. We provide a discussion of the obtained results and a criticism of the proposed model.

Organization of the chapter:

This chapter is organized as follows. In Section 5.2, we provide a detailed review of the previous work on chord estimation. Relying on this review, we introduce our point of view on the chord estimation problem in Section 4.3. We then study several approaches to estimate the chords from the succession of chroma vectors over time using HMM in Section 4.4. In particular, we describe various configurations of the observation probabilities (Section 4.4.3) and transition probabilities (Section 4.4.4). In Section 4.5, we evaluate and compare our approach to previous models. A conclusion closes this chapter.

4.2 Previous Work on Chord Estimation

In this section, we review a number of chord estimation methods. We distinguish between approaches employing a probabilistic model (Section 4.2.2) and pattern-matching-based approaches (Section 4.2.3). We also discern some recent approaches that are devoted to real-time implementation (Section 4.2.4).

4.2.1 Extraction of Signal Features That Describe the Harmonic Content

The first stage of a chord estimation system consists in extracting some low-dimensional features from the audio signal that are appropriated to the task. Since their introduction in 1999, Pitch Class Profiles (PCP) [Fuj99] or chroma-based representation [Wak99] have become common features for estimating chords or musical keys from audio recordings. PCP/chroma vectors are low-dimensional features that represent the intensity of the twelve semitones of the pitch classes. Fujishima [Fuj99] uses the chroma representation to derive a large set of chords using either a nearest-neighbor or a weighted sum pattern matching method. 27 complex chords are considered. The system is successfully evaluated on synthetic sounds from a YAHAMA PSR-520 electronic keyboard and on a real-audio excerpt: the opening theme of Smetana's Moldau. Because the chroma features emphasize the harmonic content of the signal, most of the works on chord estimation are based on this representation.

Recently, a new feature called the Tonal Centroid has been proposed by Harte *et al.* [HSG06]. This feature can be viewed somehow as an extension of the chromagram. Lee [Lee08] uses this feature in the context of chord estimation and shows that his chord estimation system performs better than when using chroma features.

It can be noticed that other features have been explored for the chord estimation task. For instance, [Lee05] proposes a novel approach based on human perception for automatic chord estimation from the raw audio data using the Summary Autocorrelation Function as signal features. [Lee06a] introduces a feature vector called the Enhanced Pitch Class Profile (EPCP) that is based on the Harmonic Product Spectrum. These features have been investigated in order to take into account the overtones generated by the chord tones. However, chroma features have almost been exclusively used as a front end to existing chord estimation models.

4.2.2 Statistical Machine Learning Techniques for Chord Estimation

4.2.2.1 HMM-based Baseline Approaches

Raphael [Rap02] uses HMMs trained by the Expectation Maximization (EM) algorithm to transcribe piano music in terms of chord labels. The final purpose of his work is a piano MIDI transcription. The chord dictionary thus distinguishes between each different combination of simultaneous notes, resulting in a very huge state space. The model is trained on various Mozart piano sonata movements and evaluated on clean recordings of solo piano music. Results on a performance of the 3rd movement of the Mozart piano Sonata K. 570 are reported.

The first system evaluated on rich polyphonic music recordings (whole pieces of music of commercial recordings) is presented by Sheh & Ellis in [SE03]. They show that chromagram features outperform cepstral coefficients for the purpose of chord estimation of real-world musical recordings. Their system draws on the prior work of [Rap02]. However,

rather than considering every possible note combination, they use a reduced set of 147 chords, having a single model for each chord type. The chord lexicon is composed of 7 chord families (maj, min, maj7, min7, dom7, aug, dim) and 21 roots (A, B, C, D, E, F, G, Ab, Bb, ..., Gb, A#, B#, ..., G#). The sequence of chord names (without chord boundaries) is used as an input to the model. Both the model parameters for chords and for chord transitions are unsupervisedly learned from flat start initializations using the forward-backward algorithm. The Viterbi algorithm is used for forced alignment or chord label recognition. The system is trained and evaluated on a small collection of 20 early Beatles songs. Considering the rather large number of chords and the small amount of training data, the chord recognition accuracy is poor. However, this work initiated the use of HMM-based approaches for the purpose of audio chord estimation. Since then, the HMM approach for chord estimation has been followed by many other researchers.

In the context of automatic structure detection for popular music, Maddage *et al.* [MXKS04] [Mad06] employ a similar learning method for chord estimation using a HMM. However, the chord model is different from [SE03]: 48 HMMs are used to model 12 major, 12 minor, 12 diminished and 12 augmented triads. Each model has five states, including entry, exit and three Gaussian mixtures (GM) for each hidden state. The mixture weights, means and covariances of all GMMs, as well as the initial and transition state probabilities are computed using the Baum-Welch algorithm. The Viterbi algorithm is then applied and gives a first estimation of the chord progression. A post-processing step is incorporated to correct possible misclassifications. Key determination is performed so that chords not in the detected key are disallowed and replaced by other chords with high probability or with the previous chord. Time alignment of the chords is corrected using heuristics derived from popular music composition knowledge. The model is trained with real songs and additional synthesized audio chord samples. Cross-validation experiments on 40 popular music songs in [MXKS04] (50 in [Mad06]) show that the chord estimation results are improved thanks to the music knowledge-based post-processing step.

Bello & Pickens [BP05] improve the approach proposed by [SE03] by encoding musical knowledge into the model. The feature extraction part is ameliorated in one part by a tuning stage [HS05] and in the other part by the use of beat (tactus)-synchronous features that minimizes the effect of local variations and transients. Finally, the chord lexicon is limited to the 24 major and minor triads since the purpose of the work is to achieve a robust mid-level representation that describes the harmonic character of an input signal, rather than an academic chord transcription from audio. The chroma features are used as observations on a 24-state hidden Markov model, where each state corresponds to one of the major and minor triads. The observation distribution is modeled by a single Gaussian. The parameters of the model are initialized using simple musical knowledge about the key distance in a circle of fifths. The model is then selectively trained in an unsupervised fashion using the Expectation-Maximization (EM) algorithm, assuming that a chord template or distribution is almost universal regardless of the type of music and thus disallowing adjustment of distribution parameters. The chord progression is obtained by decoding the model using the Viterbi algorithm. The model is tested on two early Beatles albums, *Please Please Me* and *Beatles For Sale*. Experiments show that the use of musical knowledge is crucial, that selective training introduces substantial gains into the approach and that the use of a tactus-based feature set clearly outperforms the frame-by-frame

estimation.

This last result is also claimed by Maddage *et al.* [MKL06] who propose a hierarchical approach to model the tonal characteristics of musical audio. Major, minor, diminished and augmented chords are considered. The usual pitch class profile (PCP) features are compared to psycho-acoustical profile (PAP) features that are presented as the expansion of PCP features. They consider effects of the notes in all the octaves individually. Evaluation on 40 English songs (10 Michael Learns To Rock, 10 Bryan Adams, 6 Beatles, 8 Westlife and 6 Backstreet Boys) shows that the effects of f0, sub-harmonic, and harmonic of the notes which comprise a given chord, are important clues for chord detection. It is found that the best features are the PCP where note effects (f0, sub-harmonic, harmonic) are averaged across the octaves. It is also found that tonal characteristics are better extracted using tempo proportional signal segmentation than using fixed length segmentation.

Ryynänen & Klapuri [RK08b] present a method for chord estimation related to [BP05] in the sense that it uses a chord HMM where the states correspond to the major and minor triads. The proposed chord estimation method is only one part of a global model that attempts to provide a useful representation of polyphonic popular music songs. The purpose is the automatic transcription of the chord progression, the bass line and the key signature of audio files. In the front end, the system extracts two frame-wise features: a pitch-salience estimator and an accent estimator that indicates potential note onsets based on signal energy. The chord transcription method uses a 24-state HMM where the observation likelihoods are obtained by mapping the pitch salience into a pitch-class representation, and comparing them with trained profiles for major and minor chords. Two PCPs are used, one for low-register MIDI notes 26-49 and one for high-register MIDI notes 50-73. Chord transition probabilities are estimated from training and the chord progression is found using the Viterbi decoding algorithm. The method is evaluated using a two-fold cross validation on 8 Beatles albums.

4.2.2.2 Simultaneous Estimation of Chords and Musical Context

Some HMM-based chord recognition systems use context information to improve the chord progression. Additional musical attributes (such as key, meter or structure) may be modeled simultaneously with the chords.

Lee & Slaney [LS08] follow an HMM-based approach for chord extraction similar to [SE03] and [BP05] in that the states in the HMM represent chord types and that the most probable chord sequence is found in a maximum-likelihood sense. However, they use the tonal centroid feature instead of the chroma feature. Moreover, the parameters of the model are supervisedly learned without using an EM algorithm, but directly from labeled training data. Symbolic data are used to automatically obtain a large set of labeled training data, avoiding the tedious task of human annotation of chord names and boundaries. The large amount of training data allows the building of key-specific HMMs, which not only increase the chord estimation accuracy but also provide key information. The model is evaluated on two pieces of classical music, Bach's keyboard piece Prelude

in C Major and Haydn's string quartet Op.3, No.5: Andante measures 1-46, and on two Beatles albums, *Please Please Me* and *Beatles For Sale*. Experimental results show that the approach compares favorably to the state-of-the-art [BP05]. The tonal centroid feature is found to outperform the conventional chroma feature. Chord accuracy results are improved considering musical key information.

Burgoyne & Saul [BS05] also present an HMM-based model that tracks key simultaneously with chords. It is claimed that transitions between chords are dependent of their tonal context. On the contrary to [LS08], they do not assume that music remains in a single key from start to end. The model considers chord and key to be inseparable properties of any given harmony. The model is restricted to major and minor triads. Each state of the HMM represents a chord in a possible key (C major in the key of A minor for instance). Simplified rules of tonal harmony are encoded in the transition matrix. The traditional Gaussian emission distribution is replaced with a Dirichlet distribution. The model is unsupervisedly trained with the EM algorithm on five Mozart symphonies (K. 134, K. 162, K. 181, K.182 and K.183) and tested on the Minuet of Mozart symphony K. 550. The results reveal that a more advanced harmonic model is needed to improve the results.

Papadopoulos & Peeters [PP08b] present a method for simultaneously estimating the chord progression and the downbeats from an audio file. A specific topology of hidden Markov models that enables to model chords dependency on metrical structure is proposed. Each state is defined as an occurrence of a chord at a "position in the measure". The model relies on the idea that chords are more likely to change at the beginning of measures than on other beat positions in the measure. In this model, the chord progression benefits from the knowledge of the downbeats positions and conversely the downbeats are estimated relying on the chord progression. The model is evaluated on a test-set of 66 popular music songs from the Beatles and shows improvement over the state of the art. The model is further extended in [PP10] to more complex cases that include pieces with complex metric structures such as beat addition, beat deletion or changes in the meter.

The work of Mauch & Dixon [MD10] is also concerned with the simultaneous estimation of chords and other musical attributes. A 6-layered dynamic Bayesian network models jointly key, metric position, chord and bass pitch class. The most probable sequence is inferred from the beat-synchronous bass and treble chromagrams of the whole song. The model distinguishes between 109 different chords (7 chord classes in root position: maj, min, dim, aug, maj7, maj6, dom7, plus 24 major chords in 1^{st} and 2^{nd} inversion, plus one no chord "N" chords) and is evaluated on 176 audio tracks from the MIREX 2008 Chord recognition test-set.

In [MD08], Mauch & Dixon present a new approach for chord labeling in which a chord is modeled as a mixture of different sonorities. A melody range and a bass range chromagram are separately computed and simultaneously used as observations in a hidden Markov model. A sophisticated state duration modeling is proposed, in which chord durations are gamma-distributed. The system also includes a bass model. In this work, 6 chord classes are considered (major, minor, dominant, diminished, suspended, and no chord). The model is evaluated using a five-fold cross-validation procedure on 175 Beatles songs. It is shown that the new duration modeling retains the level of accuracy while it

reduces fragmentation.

The work of Mauch *et al.* [MND09] also deals with the concept of unified music analysis. The baseline is the chord estimation method proposed in [MD10]. They propose to improve the chord progression estimation by exploiting the repetitive structure of songs. They rely on the idea that the chord sequence is the same in all sections of the same type (such as chorus or verse). They thus assign the same chord progression to repeated sections. Four types of chords are considered: major, minor, diminished, dominant or no chord. The evaluation of the method on 125 Beatles songs shows improvement in chord accuracy and reveals that the chord transcription is more consistent with the repetitive structure of the song.

4.2.2.3 Introducing Language Modeling, N-grams

Some approaches for chord estimation employ *Language Modeling* (LM) because sequences of chord labels can be viewed as word sequences in natural language. The previously presented HMM-based works make the Markovian assumption that each chord symbol depends only on the preceding one, which is a simplifying assumption. Higher order probabilistic N-grams are an interesting alternative to HMMs (that correspond to probabilistic 2-grams) because they can efficiently model the actual complexity of music.

Cheng *et al.* [CYL+08] claim that the information of two adjacent chords is insufficient for recognizing longer chord sequences. They thus propose to incorporate a N-gram model that learns the common rule of chord progression into a HMM framework for chord estimation. Applications to music classification and retrieval are investigated. Two new chord features are proposed: the longest common chord subsequence and the histogram statistics of chords. Experiments on the previously cited two early Beatles albums indicate that the N-gram-based approach outperforms the typical HMM-based approach.

Scholz *et al.* [SVB08] focus on two possible limitations of N-gram-based chord estimation models: the problem of overfitting and the problem of using a single chord labeling scheme. In order to overcome these limitations, they investigate several model smoothing and selection techniques for modeling the chord sequence of a piece of music using probabilistic N-grams. Several chord labeling schemes are considered. The various configurations of the model are tested on 180 Beatles songs. The results show that the accuracy of N-grams is increased by the proposed techniques. They also show that it is possible to accurately model more complex chord types than the usual minor/major chords.

The approach to chord estimation from audio proposed by Khadkevich & Omologo in [KO09] is based on a trained HMM combined with a Language Model. Pitch Class Profile vectors are used as input to the model. The method differs from most existing approaches in the sense that a chord is not represented as a hidden state in one ergodic HMM, but a separate left-to-right HMM is created for each chord. For a given analyzed song, the most likely chord sequence is obtained using the Viterbi decoding algorithm. The resulting chord lattice is then rescored by applying a language model of high orders (3-gram, 4-gram). The model is evaluated on 175 Beatles songs using a 5-fold cross-validation procedure. Factored and standard language models are compared and it is found that the

use of a factored LM results in a small increase in performance.

The work of Schuller *et al.* [SHAR09] is related to [KO09]. It shows that incorporating a Musicological Model (MM) in an HMM-based approach for chord labeling allows improving chord accuracy. Temporal harmonic structure is incorporated by using one "Chroma Energy Distribution Normalized Statistics" (CENS) feature [MKC05] per bar. The model is trained on $19,025$ chord lead sheets[1]. It is compared to a Cross-Correlation (CC) with templates method and a Support Vector Machines (SVM) method. Experiments on a database of 100 pieces of pop and rock music that have been annotated by trained musicians[2] are conducted. The results indicate that data-driven approaches are superior to template-based approaches and that language modeling improves chord estimation.

4.2.2.4 Other Statistical Modeling Approaches

Machine-learning-based methods for chord estimation also include approaches other than HMMs. For instance Paiement *et al.* [PEBB05] present a graphical probabilistic model where contextual information related to the meter is used to model the chord progression in order to generate chords. The graphical model uses probabilities of chord substitutions that are derived from a continuous distributed representation for chords. In this distribution, perceptually similar chords tend to be close in Euclidean distance. In the graphical models, the parameters are learnt with the EM algorithm and the Junction Tree algorithm is used for inference. The model is validated using 52 jazz standard excerpts from Sher (1988) [She88] interpreted and recorded by one of the authors in MIDI format on a Yamaha Disklavier piano. Experiments show that chord progression dependencies to the meter can be better captured with a tree structure rather than with a HMM.

The use of HMMs is compared to the use of conditional random fields (CRFs) by Burgoyne *et al.* [BPKF07]. Audio is modeled with PCP features and various configurations of HMMs and CRFs models are implemented. Cross-validation and comparison of the systems is conducted on the same set of Beatles songs than Sheh & Ellis [SE03]. It is demonstrated that the CRF-based method yields to results close to the ones obtained with the best HMM-based method, while using much fewer model parameters.

Other statistically-based chord estimation approaches include hypothesis-search-based methods. Yoshioka *et al.* [YKK+04] propose a method that concurrently recognizes chord boundaries, chord symbols and keys. This approach allows taking into account the mutual dependency of chord-boundary detection and chord-symbol identification as well as the mutual dependency of chord-symbol identification and key identification. The core of this algorithm is a hypothesis-search algorithm that evaluates tuples of chord symbols and chord boundaries. Three criteria are taken into account: acoustic features, chord progression patterns and bass sounds. Likely hypotheses are followed while highly unlikely hypotheses are pruned after a while. At the end of the song, the most probable path is chosen as the chord progression. The accuracy of the chord transcription, measured on one-

[1] "The on-line guitar archive," in http://www.olga.net, 2006.

[2] The list of the songs can be found at: "Songlist chord data-set," in http://www.mmk.ei.tum.de/sch/chord.txt, 2006.

minute excerpts from seven songs of RWC-MDBP-2001 [GHNO02] (No.14, 17, 40, 44, 45, 46, and 74), is improved considering chord progression patterns and bass sounds. However, the correctness is not improved because the proposed method makes many insertion errors.

Although information about bass sounds is used in [YKK+04], it is not integrated into a probabilistic framework. Errors in estimating bass tones tend to produce errors in the chord estimation. Sumi *et al.* [SIY+08] improve the hypothesis-search-based method proposed in [YKK+04] by probabilistically integrating bass pitch estimation into the model to improve chord estimation. Evaluation of the proposed methods on 150 one-minute excerpts of Beatles songs shows that the baseline method has been improved.

4.2.3 Pattern Matching Approaches

Alternative to the machine learning approaches for chord estimation are the pattern matching approaches. In such approaches, each feature vector computed from the audio signal is correlated with a set of chord templates that indicate the perceptual importance of the notes within a chord. The estimated chord is obtained by selecting the template that gives the maximum correlation coefficient.

Harte & Sandler [HS05] estimate chords by comparing predefined chord templates that are simple bit masks[3] to chroma features. The originality of their work is that it proposes a tuning algorithm to accurately locate the boundaries between semitones. This allows the calculation of a novel semitone-quantized chromagram. The model can distinguish between 48 chords. The model is evaluated on two early Beatles albums, *Please Please Me* and *Beatles For Sale*.

Oudre *et al.* [OGF09a] propose a chroma, template-based method for chord recognition. They rely on the idea that in a given chroma vector corresponding to a chord, the amplitudes of the notes that comprise the chord should be larger than the ones of the non-played tones. They investigate the influence of several parameters in the model. They examine several chord templates that take into account one or more harmonics for the notes, as previously proposed in [PP07]. They compare the use of several measures of fit between the chroma features and the chord templates. They also explore the influence of the number and the types of the chords that are considered in the model. Performance of the system is evaluated on 13 Beatles albums.

Some template-based approaches include post-processing steps to correct chord estimation errors. Shenoy *et al.* [SMW04] propose a symbolic inference-based chord estimation method. Individual notes are identified from beat-synchronous chroma features by considering only the elements with the four highest values in the chroma vectors. Symbolic inference is used to determine major and minor chords. The chord estimation accuracy is not sufficient to provide a usable chord transcription. This method is improved by Shenoy & Wang in [SW05] where a post-processing step similar to the one in [Mad06] is proposed. Three rule-based chord accuracy enhancement steps based on musical key and meter infor-

[3] A bit mask is a 12-dimensional vector corresponding to the 12 semitones of the pitch classes with 1 when the note belongs to the chord, 0 otherwise.

mation are used. Firstly, chords that do not belong to the key of the song (assumed to be constant over time) are eliminated. Secondly, the chord progression is smoothed so that if a chord is different from two same adjacent chords, it is replaced to match the adjacent chords. Finally, chord changes are favored at the beginning of the measures instead of other half-note time. Experiments are performed on 30 popular English songs and show that the chord estimation accuracy is spectacularly improved by the post-processing steps based on music knowledge, increasing from a relatively low score of 48.13% to a score of 78.91%.

Reinhard *et al.* [RSN08] also introduce an approach to improve chord estimation accuracy. A post-processing step to chord estimation algorithms is proposed to correct possible misclassifications caused for instance by the presence of percussive sounds or harmonics. The method is based on musical harmony principles. It works with a probability-based classifier that is solely based on the chromagram feature extracted in the previous step and that exploits the knowledge about the distribution in the neighborhood of a chord. The main assumption is that a chord is more likely to be from a pool of chords in the neighborhood, than to be any other arbitrary chord. The classifier does not only predict the most probable chord, but also returns a probability of confidence for every possible chord considering the observed chromagram. Three different classifiers (scalar product pattern matching, Mahalanobis distance classifier, Naive Bayes classifier) are used in order to demonstrate that the proposed post-processing technique can be used in combination with arbitrary classifiers.

As in machine learning approaches, some template-based approaches are also based on music theory. For instance the purpose of Zenz & Rauber [ZR07] is to incorporate music theoretical knowledge in a chord extraction algorithm without restricting the input data to a narrow range of musical styles. The algorithm distinguishes between major, minor and diminished chords. This work uses Pitch Class Profile features computed on beat-synchronous frames using the Enhanced Autocorrelation (EAC) Algorithm [TK00]. The generated PCPs are compared to a set of reference chord PCPs that are empirically determined from one-minute excerpts of 5 popular songs. A single key is estimated for each song and key information is used to refine the set of possible chords. The context of each chord is analyzed for estimating the final chord progression. Evaluation is performed on a set of 35 pieces of various music styles and indicates that music theory information improves chord estimation accuracy.

4.2.4 Real-Time Implementation for Chord Estimation

Some recent works are concerned with real-time implementation of chord estimation methods. Cho & Bello [CB09] propose a real-time implementation of HMM-based chord estimation based on the model proposed in [BP05]. To overcome the limits of the online processing (limited memory capabilities and no access to future observations), they propose a system of buffers. Modifications are introduced in the standard Viterbi decoding algorithm to approximate offline results while minimizing the system's latency. 12-fold

cross evaluation on the MIREX 2008 169 Beatles songs show that the results of realtime decoding converge towards the non-realtime decoding result.

Stark & Plumbley [SP09] propose a real-time chord recognition system using a classification technique based on residual energy in the chromagram. They develop a chromagram calculation method in which unwanted energy such as noise is discarded. Experiments are carried on a set of 180 chord samples extracted from real-world guitar recordings. 108 different chords are considered (the 12 variations of major, minor, diminished, augmented, suspended 2^{nd}, suspended 4^{th}, major 7^{th}, minor 7^{th} and dominant 7^{th} chords). The proposed chroma computation method is shown to compare favorably with other state-of-the-art methods [BP05] [SE03].

Konoki *et al.* [KM10] describe a system that estimates in real-time chord labels from sounds generated by electric guitars. Two difficulties related to chord estimation are addressed: "omitting", "inversions" and "tension voicing" notes as well as enharmonic equivalence. The system starts by computing chroma vectors from which the theoretically played notes are estimated. For this, the four highest strong pitch classes that have an intensity above a threshold are selected. Possible chord labels are then listed by using a "search tree". The model is evaluated in real-time using guitar chords generated by a guitar player. 16 chord types are considered. The chord types employed in this study are the sixteen patterns frequently used in chord guitar performances $(maj, min, 7^{th}, m7, M7, mM7, aug, dim, 6^{th}, m6, sus4, 7sus4, 7(b5), aug7, dim7, andadd9)$. Ambiguous cases (such as enharmonic equivalence) are resolved by comparing the possible chord progressions obtained from the chord labels with some chord progression patterns extracted from a "chord progression database".

4.2.5 Summary of Chords Estimation Techniques

4.2.5.1 Summary of the Above-Presented Methods

Tables 4.1, 4.2 and 4.3 list the characteristic attributes of the above-presented chord estimation methods. The systems are presented in the chronological order. The column "Method" indicates the main techniques that are used for chord estimation. The column "Input features" indicates the type of input that are processed. The column "Comments" underlines some interesting specific strategies that are adopted. The column "Chord lexicon" indicates the chords lexicon that can be handled by the systems. Finally, the column "Evaluation material" indicates the musical material on which the systems have been tested and possibly trained.

4.2.5.2 Summary of the MIREX Chord Recognition Systems

In this section, we present an overview of the chord estimation algorithms submitted to the MIREX 2008 and 2009 contests.

• Introduction to MIREX Chord Recognition Task

The first audio chord detection task in Music Information Retrieval Evaluation eX-

Table 4.1: Characteristics of some chord estimation methods 1999-2006.

Reference	Method	Input features	Comments	Chord lexicon	Evaluation material
Fujishima [Fuj99]	Pattern Matching	PCP	• nearest neighbor or a weighted sum method • bit-mask chord templates	27 complex chords	• no training • testing on synthetic sounds + one real-audio excerpt (50s)
Raphael [Rap02]	HMM	collection of features	unsupervised training by EM	chord considered as any combination of simultaneous notes	• training on various Mozart piano sonata movements • testing on the 3^{rd} movement of Mozart piano Sonata 18, K. 570.
Sheh & Ellis [SE03]	HMM	PCP	• unsupervised learning with EM • random initialization • Viterbi decoding for forced alignment or chord label recognition	147 complex chords	• training on 18 early Beatles songs • testing on 2 Beatles songs
Maddage *et al.* [MXKS04], [Mad06]	HMM	PCP	• 48 HMMs, one for each chord, 3 states per chord • supervised training EM • post-processing step base on key and meter	48 (maj, min, dim aug)	• training: real songs + synthesized audio chord samples • Cross-validation on 40 popular music songs
Yoshioka *et al.* [YKK+04]	hypothesis-search algorithm	beat-synchronous PCP	• concurrent recognition chord boundaries, chord symbols and keys • generation of hypotheses about tuples of chord symbols and chord boundaries • 3 criteria taken into account: acoustic features, chord progression patterns and bass sounds	48 (maj, min, dim, aug)	• training: 2592 audio samples of each chord played on a MIDI tone generator + 6 RWC songs (2-fold cross-validation) • testing: one-minute excerpts from seven songs of RWC (No.14, 17, 40, 44, 45, 46, and 74)
Bello & Pickens [BP05]	HMM	beat-synchronous PCP	• musical knowledge encoded into the model • unsupervised selective training EM	24 (maj, min)	2 Beatles albums, *Please Please Me* and *Beatles For Sale*
Burgoyne & Saul [BS05]	HMM	PCP	• simultaneous keys and chords estimation• Simplified rules of tonal harmony encoded in the transition matrix • Dirichlet distribution unsupervisedly trained with EM	24 (maj, min)	• training: 5 Mozart symphonies (K. 134, K. 162, K. 181, K.182 and K.183) • testing: Mozart Symphony K. 550, Minuet
Harte & Sandler [HS05]	template-matching	PCP	• quantized chromagram • bit-mask chord templates	48 (maj, min, dim aug)	• no training • 2 Beatles albums, *Please Please Me* and *Beatles For Sale.*
Paiement *et al.* [PEBB05]	graphical model	MIDI	• contextual information related to the meter used to model the chord progression • comparison tree structure/HMM	any group of observed notes forming a chord	52 jazz standards excerpts
Shenoy & Wang [SW05]	symbolic inference	beat-synchronous PCP	• 3 rule-based chord accuracy enhancement steps based on musical key and meter information	24 (maj, min)	• no training • testing:30 popular English song
Maddage *et al.* [MKL06]	hierarchical model	beat-synchronous PCP/PAP	incorporate the note effects (F0, sub-harmonic, harmonic)	48 (maj, min, dim, aug)	• synthetically generated music chords • 40 English songs

Table 4.2: Characteristics of some chord estimation methods 2007-2008.

Reference	Method	Input features	Comments	Chord lexicon	Evaluation material
Burgoyne *et al.* [BPKF07]	CRF	PCP	Dirichlet for modeling PCP distribution	48 (maj, min, dim, aug)	10-fold cross validation on 20 Beatles songs (18 for training, 2 for testing)
Lee & Slaney [LS08]	key-specific HMM	tonal centroid	• supervised training with EM • training files generated from symbolic data	24 (maj, min) or 36 (maj, min, dim)	• training: 765 classical music files +158 Beatles songs • testing: Bach Prelude in CM and Haydn string quartet Op.3, No.5, measures 1-46 + 2 Beatles albums, *Please Please Me* and *Beatles For Sale*.
Zenz & Rauber [ZR07]	template matching	beat-synchronous PCP	• empirically-based reference PCP from one-minute excerpts of 5 popular songs • encode music theoretical knowledge about key	36 (maj, min, dim)	• no training • testing: 35 pieces of various music styles
Cheng *et al.* [CYL+08]	HMM + N-grams	PCP	• Language Modeling • observation probabilities based on chord templates • 2 new chord features: the longest common chord subsequence and the histogram statistics of chords	24 (maj, min)	• training: 152 Beatles songs • testing: 2 Beatles albums, *Please Please Me* and *Beatles For Sale*
Mauch & Dixon [MD08]	HMM	melody range + bass range chromagram	• sophisticated state duration modeling • bass model	6 chord classes (maj, min, dom, dim, sus, no chord)	5-fold cross-validation,175 Beatles songs
Papadopoulos & Peeters [PP08b]	double-states HMM	beat-synchronous PCP	• simultaneous estimation chords and downbeats • observation probabilities based on chord templates + harmonics	24 (maj, min)	• no training • testing: 66 Beatles songs
Reinhard *et al.* [RSN08]	Classifier (Scalar product, Mahalanobis distance, Naive Bayes)	beat-synchronous PCP	post-processing step based on chord neighborhood	24 (maj, min)	• training: 2 Beatles albums, *Please Please Me* and *A Hard Day's Night* • testing: 2 Beatles albums, *Please Please Me* and *Beatles For Sale*
Ryynänen & Klapuri [RK08b]	HMM	multi-f0 PCP	• observation likelihoods obtained by comparison with trained profiles • 2 chromagrams are used (one for low and one for high-register)	24 (maj, min)	2-fold cross-validation, first 8 Beatles albums
Scholz *et al.* [SVB08]	N-gram	chord labels	use model smoothing and selection techniques initially designed for spoken language modeling	various labeling schemes: • Maj/min • Shorthand types • Harte's with enharmonic equivalence	13-fold cross-validation on 13 Beatles albums
Sumi *et al.* [SIY+08]	hypothesis-search	beat-synchronous PCP	interrelationship between bass lines and chords	48 (maj, min, dim, aug)	5-fold cross-validation on 175 Beatles songs

Table 4.3: Characteristics of some chord estimation methods 2009-2010.

Reference	Method	Input features	Comments	Chord lexicon	Evaluation material
Cho & Bello [CB09]	HMM + real-time decoding	PCP	• real-time processing • system of buffers • modified Viterbi decoding for real-time	24 (maj, min)	12-fold cross evaluation on the MIREX 2008 169 Beatles songs
Khadkevich & Omologo [KO09]	HMM + LM	PCP (HMM), beat-synchronous chord symbols (LM)	• a separate left-to-right HMM for each chord	24 (maj, min)	5-fold cross-validation on 175 Beatles songs
Mauch *et al.* [MND09]	dynamic Bayesian network + musical structure	beat-synchronous treble and bass chromagrams	• same chord progression to repeated sections	48 (maj, min, dim, dom) + no chord.	5-fold cross-validation on 125 Beatles songs
Oudre *et al.* [OGF09a]	Pattern Matching	PCP	• investigate various measures of fit • study chord type influence	4 chord classes (maj, min, dom7, min7)	• no training • testing: 13 Beatles albums
Schuller *et al.* [SHAR09]	HMM + MM	one CENS per bar	• learn typical chord successions with musicological model • comparison data-driven/template-based approaches	• 24 (maj, min)• 36 (maj, min, and "other")	• training: 19,025 chord lead sheets • testing: 100 pieces of pop and rock music
Stark & Plumbley [SP09]	frame-based classifier for real-time use	PCP	• classification based upon chroma residual energy • allows for inharmonicity in signal	108 (maj, min, dim, aug, sus2, sus4, maj7, min7, dom)	• no training • testing: 180 chord guitar audio samples
Konoki *et al.* [KM10]	search tree	PCP	• "omitting", "inversions" and "tension voicing" notes • enharmonic equivalence	16 chord classes	• no training • testing: guitar sounds
Mauch & Dixon [MD10]	dynamic Bayesian network	bass and treble chromagrams	• simultaneous estimation chords and musical context	109 complex chords	• no training • testing: 176 Beatles songs.
Papadopoulos & Peeters [PP10]	double-state HMM	beat-synchronous PCP	• simultaneous estimation chords and downbeats in variable meter	24 (maj, min)	• no training • testing: 169 Beatles songs

change[4] was organized in 2008. The MIREX 2008 Audio Chord Detection task was divided into two subtasks. In the first subtask the systems were pre-trained and tested against 176 Beatles songs. In the second subtask systems were trained on 2/3 of the Beatles test-set and tested on 1/3. An overlap score was calculated as the ratio between the overlap of the ground truth and detected chords and ground truth duration. Four songs were excluded from the original Beatles test-set because of problems when aligning the ground truth chords to the audio data.

The MIREX 2009 audio chord detection[5] task description is similar to the one proposed in 2008 except that the score computation is slightly different. A first score is calculated as the ratio between the overlap of the ground truth and detected chords and ground truth duration, then a weighted average is computed across the songs by weighting each score by the song duration. In 2009, the test-set also included 37 popular music songs. A total number of 13 algorithms were submitted to the pre-trained systems subtask, and 5 algorithms were submitted to the trained systems subtask.

• **Methods and Results**

Tables 4.4 and 4.5 give a brief description of the various algorithms submitted to MIREX 2008 and MIREX 2009 Audio Chord Detection task.

Figures 4.1 and 4.2 indicate the chord accuracy results obtained by the various algorithms submitted to the MIREX 2008 and MIREX 2009 Audio Chord Detection task.

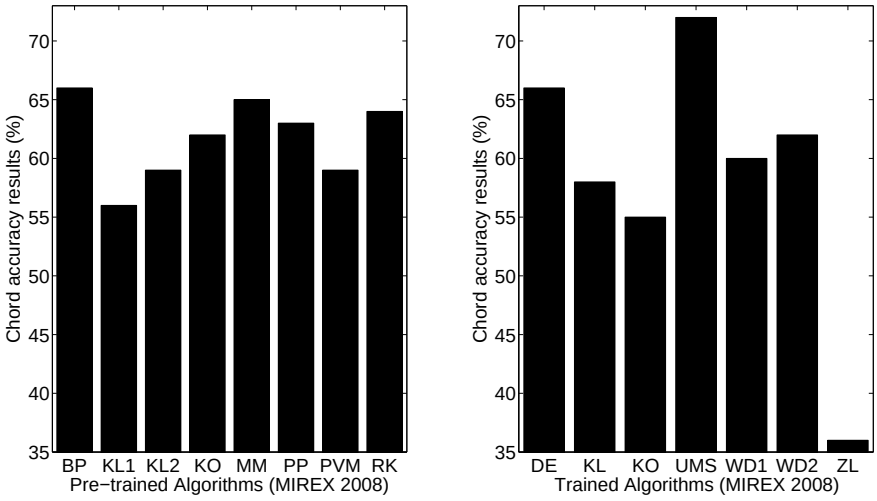

Figure 4.1: MIREX 2008 Audio chord detection results (in %) for the pre-trained systems (left) and for the trained systems (right).

[4]http://www.music-ir.org/mirex/2008/
[5]http://www.music-ir.org/mirex/2009/

Table 4.4: Summary of audio chord detection systems submitted to MIREX 2008.

Reference	Method	Input features	Training	Comments
Bello & Pickens BP	HMM	beat-synchronous CQT-PCP	yes	tuning, music knowledge
Ellis DE	HMM	beat-synchronous IF-PCP	yes	tuning, uses 2 chromagrams (including one to emphasize the bass line)
Khadkevich & Omologo KO	HMM	DFT-PCP	yes	one separate HMM for each chord, 512 12-dimensional GM
Lee KL, KL1 & KL2	HMM	tonal centroid	yes	key-specific HMM
Mehnert *et al.* MM	HMM	chromagram mapped to circular pitch spaces (CPS) [GMAB08]	yes	Symmetry Model used as basis for the chord analysis system
Papadopoulos & Peeters PP	HMM	DFT-PCP	no	tuning, music knowledge and chord templates considering harmonics
Pauwels *et al.* PVM	probabilistic framework based on Lerdahl's tonal distance metric	multi-f0s PCP	no	simultaneous chords/keys
Ryynänen & Klapuri RK	HMM	multi-f0s PCP	yes	2 chromagrams (low and high registers)
Uchiyama *et al.* UMS	HMM	PCP	yes	Harmonic/Percussive sound separation front-end
Weil & Durrieu WD1 & WD2	HMM	CQT-tonal Centroid	yes	tuning, attenuation of the main melody
Zhang & Lash ZL	HMM	DFT-PCP	yes	pre-processing step which detects silences

Table 4.5: Summary of audio chord detection systems submitted to MIREX 2009. IF: Instantaneous Frequency, HPCP Harmonic Pitch Class Profile, HCDF Harmonic Change Detection Function.

Reference	Method	Input features	Training	Comments
Ellis DE	HMM	beat-synchronous IF-PCP	pre-trained	tuning, key-relative transition matrix, maximal gamma values instead of Viterbi path
Harte & Sandler CH	Template matching	CQT-HPCP	no	tuning, chord boundaries based on an HCDF
Khadkevich & Omologo KO1 & KO2	HMM	beat-synchronous DFT-PCP	pre-trained	separate models are built for each chord distinguished by the system
Mauch *et al.* MD	Bayesian network	beat-synchronous note salience representation PCP	no	separate bass and treble chromagrams, structure repetitions used to improve chord estimation
Oudre *et al.* OGF1 & OGF2	template matching	CQT-PCP	no	systems 1 major & minor chords and 2: major, minor and dominant 7^{th} chords
Papadopoulos & Peeters PP	HMM	beat-synchronous CQT-PCP	no	tuning, simultaneous chords/downbeats estimation
Pauwels *et al.* PVM1	probabilistic framework based on Lerdahl's tonal distance metric	multi-f0s-PCP	no	simultaneous chords/keys
Pauwels *et al.* PVM2	template matching	multi-f0s-PCP	no	binary templates
Reed *et al.* RUSUSL	HMM	dynamic features of chroma vectors	yes	harmonic/percussion source separation, tuning, minimum classification error learning
Rocher *et al.* RRHS1 RRHS2 RRHS3	note segment	graph, rule-based, dynamic programming	no	interaction chords/key
Weller *et al.* WEJ1, WEJ2, WEJ3 and WEJ4	large margin structured prediction approach (SVM-struct)	beat-synchronous PCP	yes	MaxGamma decoding

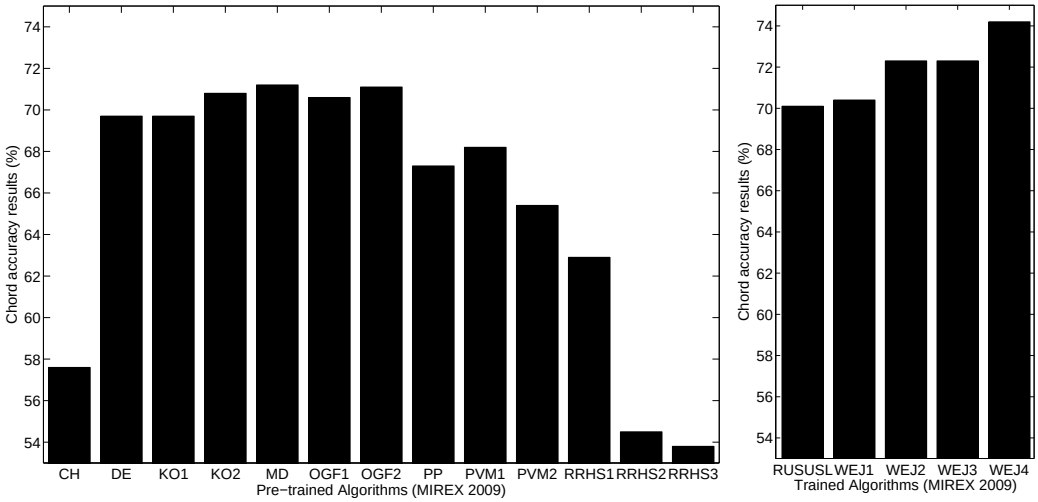

Figure 4.2: MIREX 2009 Audio chord detection results (in %) for the pre-trained systems (left) and for the trained systems (right).

4.3 Proposed Approach for Chord Estimation

As in most of the previous methods, we have chosen to use the chroma features as signal observations and represent the chord progression using a hidden Markov model. We mainly rely on the above-mentioned approach [BP05] as we incorporate musical knowledge in our model. Various ways of constructing the HMM are studied using either music theory, results from cognitive studies, smoothed training, multivariate Gaussian models or normalized-correlation. We also pay attention to the problem of taking into account the overtones produced by the musical acoustic instruments in the model.

4.3.1 Hidden Markov Models

Since their introduction in the late 1960s, the hidden Markov models (HMMs) have been widely used in many different research areas, including speech processing and more recently music information retrieval. Real world is full of processes that we wish to understand via observation. These processes produce observable outputs that can be characterized as signals. Markov models are statistical models used to describe systems from which each observation corresponds to a physical event, usually called *state*.

The hidden Markov models (HMMs) are an extension of the Markov models that are used when the states cannot be directly observed (they are hidden), but can be observed through another set of stochastic processes that produce the sequence of observations (the observation is a probabilistic function of the state). For a tutorial on hidden Markov models, we refer the reader to the work of Rabiner [Rab89].

4.3.2 On the Use of HMM for Chord Estimation

According to [Rap02], there are two major advantages when modeling the chord progression using a HMM. First of all, the realization of a given state, for instance a C major chord, depends on a wide range of parameters such as the instrumentation, the dynamics, the room acoustics etc. The realization of two CM chords produced in different conditions may result into extremely different signal observations. This variability of configuration of the data can be handled using statistical machine learning approaches.

Secondly, the structure of musical data can be captured using a probabilistic framework. The major reason why we use HMM for chord estimation relies on this second argument. We can exploit the structure of musical data (interaction between keys, chords and downbeats) using a HMM. It allows us to incorporate in a simple manner some information related to the inherent structure of Western tonal music and build rich models that are specific to music. In general, compositors take into account musical rules to create a piece of music. The harmony is related to many other musical attributes and is part of a global musical context. For instance, chord transitions follow some musical rules that can be embedded in the state transition matrix of the HMM. We will also see in Chapters 5 and 6 of this dissertation that the use of an HMM allows us to consider interaction between harmony and other musical attributes such as the meter and the key.

4.3.3 The Problem of the Harmonics

We follow most of the previous works on chord estimation and use chroma features extraction as front-end of our system. Our observations thus consist of 12-dimensional vectors that represent the intensity of the 12 semitones of the equal-tempered scale of Western tonal music.

A weakness of most of the previously proposed methods is that they operate a direct mapping between the PCP/chroma values and the pitch of a note, *i.e.* a C note is represented by a single non-zero value in the chroma vector. In other words, the assumption is made that what we observe in the spectrum is directly the pitch of the notes. As underlined in Section 3.5.1 of Chapter 3, in a spectral representation, we do not observe directly the various pitches but a mixture of their harmonics that will result in a mixture of non-zero values in the chroma vector. Therefore, values at pitch classes other than those of the notes will occur in the chroma vectors. For this reason, we propose to consider the presence of the harmonics in the parameters of the model.

In [MKL06], Maddage et al. experimentally show that the effects of $f0$, sub-harmonic and harmonic of the notes, which comprise the chord, are important for chord estimation. Some works related to chord or key estimation also focus on this problem. The presence of harmonics is taken into account either when computing the chroma features or in the model parameters. The first approach is followed for instance by Pauws [Pau04] who computes the chromagram using an auditory perception inspired front-end so that the perceptual pitch and the musical background are simultaneously taken into account.

Zhu et al. [ZKG05] extract from the constant-Q spectrum only the partials which are consonant according to a diatonic scale, using a filtering method, called consonance filtering. Peeters [Pee06a] proposes the use of a Harmonic Peak Subtraction function which reduces the influence of the higher harmonics of each pitch. Lee [Lee06a] proposes a chroma feature called the Enhanced Pitch Class Profile that takes into account the overtones generated by the chord tones. The chromagram is not directly computed from the DFT but from the Harmonic Product Spectrum. The second approach is followed for instance by Izmirli [Izm05] who measures the contribution of the harmonics on a piano database. The contribution of the harmonics of a note is taken into account in Paiement et al. [PEBB05] and Gómez [GÓ6b] using a theoretical spectral envelope. It relies on the property that the amplitude of the h^{th} harmonic $f_h = hf_0$ of a note of fundamental frequency f_0 can be modeled with geometric decaying ρ^h , with $0 < \rho < 1$.

We propose to take into account the presence of harmonics in our model for chord estimation relying on the model presented in [GÓ6b]. This model extends the Pitch Class Profiles (PCPs) to the Harmonic Pitch Class Profiles (HPCPs). For this, a theoretical amplitude is attributed to each harmonic composing the spectrum of a note with an empirical decay factor set to 0.6 in the experiments so that this contribution decreases with the frequency. The contribution for the first 6 harmonics of a note is given in Table 4.6. Therefore, higher harmonics contribute to the pitch class of their fundamental frequencies. In spite of its over-simplicity, and even if this approach provides an extremely rough approximation of the spectral envelope of musical instrument sounds, it has empirically been proved to be robust in the case of key estimation. For instance [Pee06b] has compared a template-based approach relying on the model proposed in [GÓ6b] with an HMM-based approach using a database consisting of 302 European baroque, classical and romantic music extracts. It was found that the cognitive-based approach performed better than the HMM-based approach. This is why we propose to use this approach for chord estimation purpose.

Table 4.6: First 6 harmonics of a note and given amplitudes.

n	1	2	3	4	5	6
frequency	f	2.f	3.f	4.f	5.f	6.f
factor	1	s	s^2	s^3	s^4	s^5

4.4 Chord Estimation From the Chroma Vectors Using a HMM

We describe here several methods to estimate the chord progression of an audio signal over time. All these methods are based on the hidden Markov models (HMMs) [Rab89]. The various methods differ in the way observation probabilities and transition probabilities are computed.

4.4.1 Model

4.4.1.1 Chord Lexicon

Following a large part of the previous works, we restrict our harmonic content analysis to a limited set of chords composed of the $I = 24$ major and minor triads (C major, ..., B major, C minor, ..., B minor). The notation for chord types will be the following: CM, ..., BM, for major chords, Cm, ..., Bm for minor chords. We do not make any distinction between enharmonic equivalent (C#/Db, E#/F, ...). We did not include other chords, neither simpler such as dyads, more complex such as 7^{th} chords nor diminished or augmented chords (even if this last categories of chords was considered in several previous works such as in [HS05]). We acknowledge that the harmonic progression of a piece of music cannot be fully described according to music theory with such a limited chord lexicon. However, we choose to limit our chord dictionary to the 24 major and minor triads for the following reasons:

- Firstly, we find it sufficient to describe the harmonic characteristics of a wide range of music types. Previous works on music classification have shown that this reduced set of chords is sufficient to describe the harmony content of music for similarity applications such as cover version estimation[6]. See for instance [Lee06b] or [Bel07].
- Secondly, we think that, by limiting the number of chords in the lexicon, we can avoid overfitting to a particular type of music during training.
- Moreover, a larger chord lexicon would require a larger amount of manually labeled training data (in the case of supervised training), which is an extremely tedious task, even for well-trained musicians.
- Finally, limiting our chord dictionary to the 24 major and minor triads allows us to incorporate some theoretical and experimental music knowledge in our probabilistic model. This music knowledge we rely on is specific to the 24 major and minor triads and could not have been applied to a more complex chord lexicon.

4.4.1.2 Overview of the Proposed Model

We consider an ergodic 24-states HMM, each state representing a single chord of our chord lexicon. The hidden states correspond to the different chords (CM, ..., BM, Cm, ..., Bm). The observations correspond to each signal frame represented by a 12-dimensional chroma vector. The chord progression is obtained by decoding the underlying sequence of hidden chords from the sequence of observed chroma vectors using the Viterbi decoding algorithm. Because we use an ergodic model, all possible chord transitions are allowed. State transitions obey a first-order Markov property, *i.e.*, the future is independent of the past given the present state.

[6] Cover versions consist of different performances of the same underlying piece of music performed with variations in the style, the instrumentation, the tempo, etc.

Figure 4.3 shows a simplified graph of the HMM we use for chord estimation. For clarity, only three chords are represented in the figure (CM, C#M, DM). Each state represents a chord. At each time step, the chord generates an observable chroma vector. Any chord can move to any other chord or remain the same.

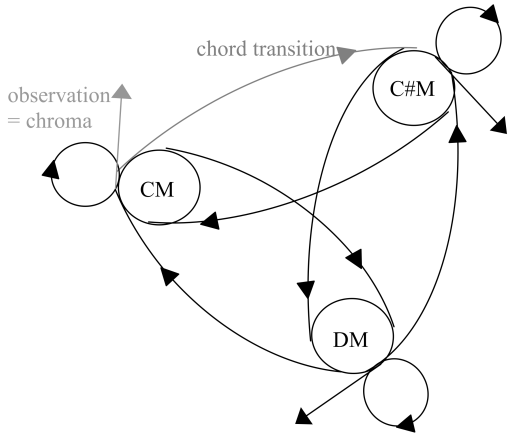

Figure 4.3: Simplified graph of the chord estimation hidden Markov model considered in this dissertation. The hidden states correspond to the chords and the observations correspond to the chroma vectors.

Each state in the model generates an observation vector, the chroma feature, with some probability. This is defined by the **observation probabilities**. In part 4.4.3, we study three approaches to define these probabilities. The first one (Method 1) learns these probabilities by training a Gaussian model on chord-normalized chroma vectors. The second one (Method 2) does not use the training set but defines probabilities based only on music theory, considering the presence of higher harmonics (using the HPCPs). The third one (Method 3) is close to Method 2 but defines probabilities based on a normalized-correlation measure rather than a Gaussian model.

In music pieces, the transitions between chords result from musical rules that should be reflected in the **state transition matrix**. This is one of the reasons why the problem is modeled using a Markov model. In part 4.4.4, we study four approaches to define the transition matrix. Method A is based on music theory: the closeness of chords in the doubly-nested circle of fifths. Method B uses the results of cognitive experiments: the closeness of chords using Krumhansl's key profiles. Method C learns the transitions probabilities from the HMM training. We finally propose a new method, D, which learns the transitions from score transcriptions.

Figure 4.4 illustrates the general flowchart of the considered model and shows the various studied configurations. In what follows, we denote by π and T, the initial state distribution and state transition probability distribution. Given the observations, we estimate the most likely chord sequence over time in a maximum likelihood sense. We now describe in detail the characteristics of our HMM: initial state distribution, observation probability distribution and state transition probability distribution.

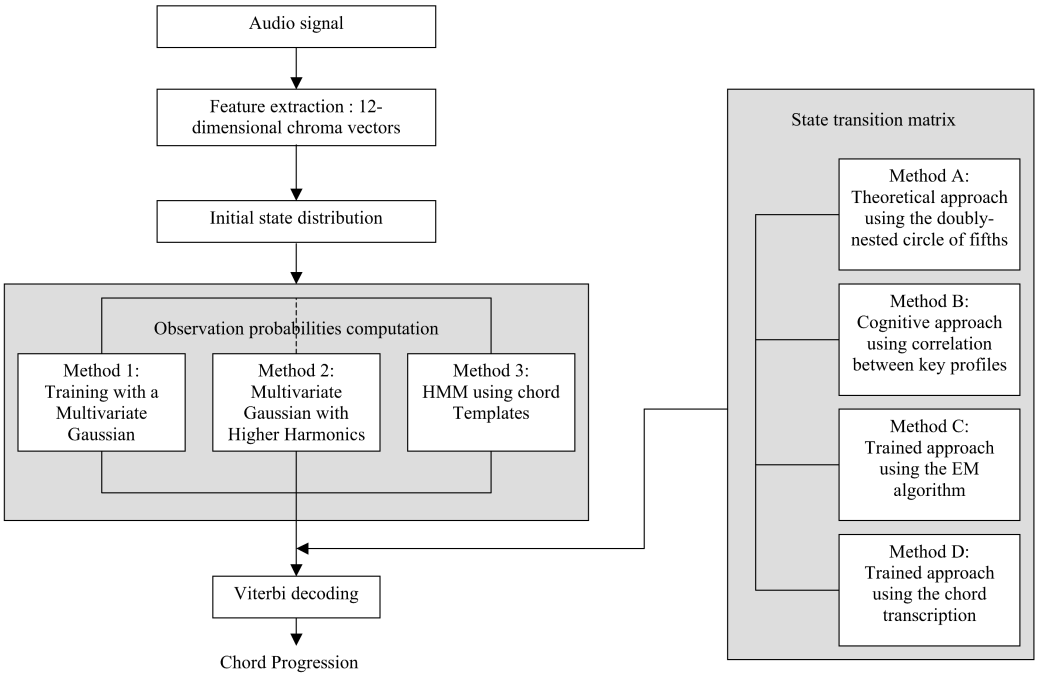

Figure 4.4: General flowchart of the studied models for chord progression estimation.

4.4.2 Initial State Distribution

The prior probability π_i for each state is the prior probability that a specific chord $i, i \in [1, 24]$ has been emitted. Since we do not know *a priori* which chord the piece begins with, we initialize π at $\frac{1}{24}$ for each of the 24 states. This choice was also taken in [BP05].

4.4.3 Observation Symbol Probability Distribution

4.4.3.1 Method 1: Modeling by a Multivariate Gaussian Trained on a Labeled Test-set

In this method, the observation distribution is modeled by 24 (one for each state) 12-dimensional single multivariate Gaussian distributions defined by their mean vectors μ_i and covariance matrices Σ_i, with i denoting the i^{th} state, $i \in [1, 24]$.

In [SE03], the model is trained using the standard expectation maximization (EM) algorithm for HMM parameters estimation. The parameters μ and Σ are initialized with random values. According to [BP05], on the one hand, the template for a chord is almost universal and should not change from song to song. On the other hand, it is unlikely that every chord of the lexicon will be present in the training test-set. This is why it is proposed to selectively train the model, disallowing adjustments of μ and Σ while π and T are updated. Experiments on 28 *Beatles* songs show that selective training results in a large increase of chord accuracy. We also believe that any reasonably sized training set will be insufficient to appropriately estimate the parameters of the model. Indeed, since

the number of observations in the test-set will likely differ among the 24 possible chords, training directly the model on the test-set may lead to overfit the model to a specific type of music (that means learning the characteristics of the test-set).

In order to learn the observation distribution for each of the 24 possible chords, we propose to first learn the model for the CM chord and the Cm chord and then map the two trained models to all possible chords by circular permutation. This allows increasing the training set of each chord type. A similar approach was proposed in [Pee06b] in the case of key estimation and in [SE03] in the case of chord estimation. We proceed as follows:

1. All the chroma vectors of the labeled training test-set are mapped to a root-note of C using circular permutation.

2. The mean vector and the covariance matrix for the CM (Cm) chord are computed from all CM (Cm) chroma vectors.

3. The mean vectors and covariance matrices for all chords are obtained from the two trained models by circular permutation.

The mean vectors for the CM and Cm chords trained on the test-set presented in Section 4.5.1 are represented in the left part of Figure 4.5. Note that in this case we do not make any assumption on the signal (instrumentation, harmonics, etc.) and we do not introduce any musical knowledge. In what follows, we will call this method "Method 1".

4.4.3.2 Method 2: Modeling by a Multivariate Gaussian Based on Music Theory Considering the Presence of Higher Harmonics

In this case, the observation distribution does not rely on any training on a given test-set. As in [BP05], the observation distribution relies directly on music theory; however a major difference with [BP05] is that we consider the presence of the higher harmonics of the theoretical notes in the construction of the multivariate Gaussian models (by modifying the parameters μ and Σ). This consideration allows us to significantly improve the results over the method proposed in [BP05].

In [BP05], the mean vectors and covariance matrices reflect musical knowledge. The mean vectors are 12-dimensional vectors with 1 if the note belongs to the chord and 0 otherwise. For instance, if we consider a 12-dimensional mean vector $\overrightarrow{\mu}$ with $\overrightarrow{\mu}(1)$ corresponding to pitch C, $\overrightarrow{\mu}(2)$ corresponding to pitch C# and so on, the mean vector corresponding to the CM chord (C-E-G) will be 100010010000 (see middle-left part of Figure 4.5).

In the covariance matrices, pitches that comprise the triad are more correlated than pitches that do not belong to the triad. The covariance between pitches that comprise the triad is thus given a non-zero value. The value is attributed with respect to music theory and empirical evidence from Krumhansl work [Kru90], that is to say that the dominant ($5th$ degree) is more important than the mediant ($3rd$ degree) in characterizing the root

of a triad [7].

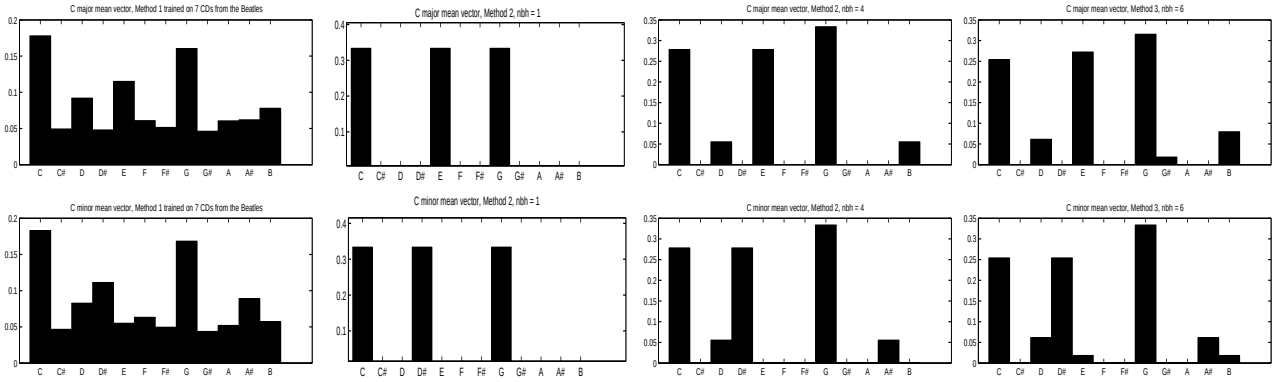

Figure 4.5: Mean chroma vectors for the C Major (upper part of each figure) and C minor (lower part) chords using [from left to right]: Method 1 (trained using 7 CDs of the Beatles), Method 2 without harmonic contribution, Method 2 with 4 harmonics contribution, Method 3 with 6 harmonics contribution (in this case, the figures represent the chroma templates instead of the mean vectors).

We now propose to take into account the contribution of the higher harmonics of the theoretical notes into the Gaussian parameters. We do this in the following way.

Mean vectors: For each note of a chord, we add the contribution of the harmonics in the mean vectors. The amplitude contribution of the h^{th} harmonic of a note is similar to the one proposed by [GÓ6b]: 0.6^{h-1}. Table 4.7 indicates the considered harmonics and the corresponding amplitudes for the CM and the Cm templates. We represent the corresponding mean vectors for CM and Cm (in the case of 4 harmonics) in the middle-right part of Figure 4.5.

Table 4.7: The first 6 harmonics and their amplitude for a CM (Cm) triad.

CM (Cm) chord						
note	**harmonics**					
C	C	C	G	C	E	G
E(Eb)	E(Eb)	E(Eb)	B(Bb)	E(Eb)	G#(G)	B(Bb)
G	G	G	D	G	B	D
amplitude	1	0.6	0.6^2	0.6^3	0.6^4	0.6^5

Covariance matrices: [BP05] only considers the correlation between the chroma vectors corresponding to the pitch of the notes belonging to a given chord. In our method, we also consider the correlation between the harmonics of each note. For example, for a CM chord (C-E-G), D is the 3^{rd} harmonic of G. Hence, we attribute a non-zero value to

[7] In [BP05], the covariance of the tonic with the dominant and of the dominant with the mediant is set to 0.8. The covariance of the tonic with the mediant is set to 0.6. Since we both use songs from the Beatles to evaluate our system, we will use the same values when testing method [BP05].

the covariance between D and G. As in [BP05], the values we use are heuristic but we still respect the rule that the dominant is more important than the mediant in characterizing the root of a triad [8]. The covariance matrices we propose for a CM and a Cm chord are represented in Figure 4.6 above the covariance matrices proposed in [BP05]. In what follows, we will call this method "Method 2".

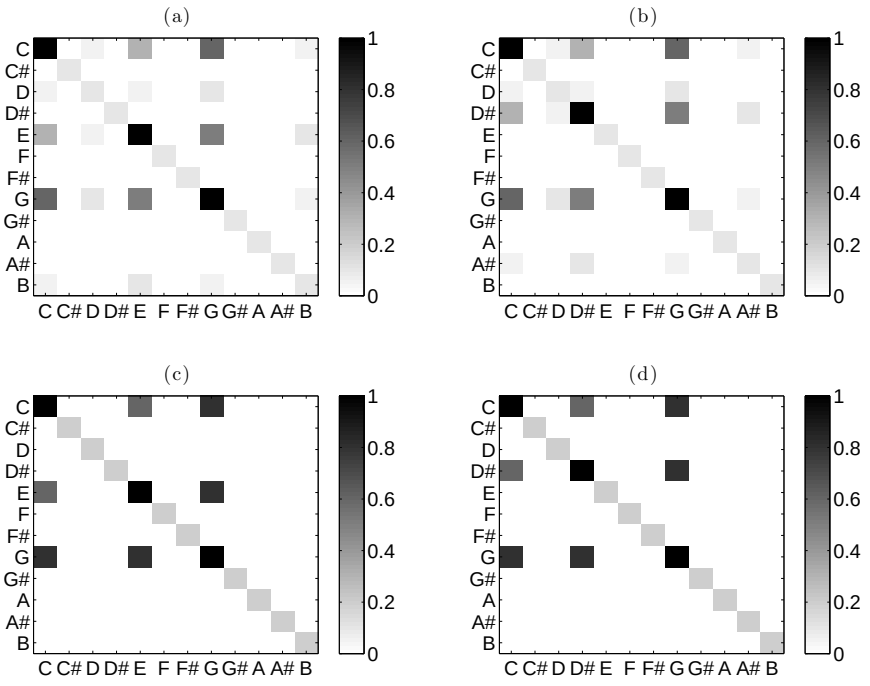

Figure 4.6: Covariance matrices for a CM (left) and a Cm (right) chord considering the presence of 4 harmonics (upper part, (a) and (b)) and proposed covariance matrices in [BP05] (bottom part, (c) and (d)).

4.4.3.3 Method 3: Probability Derived from Correlation with Chord Templates

In this method, the observation probabilities are not modeled by a multivariate Gaussian distribution. They are obtained by computing the correlation between the observation vectors and a set of chord templates.

[8] The covariance of the tonic with the dominant is set to 0.6; the covariance of the dominant with the mediant is set to 0.5; the covariance of the tonic with the mediant is set to 0.3; the covariance of a note with its second harmonic is set to 0.1; the other non-zero values are set to 0.05. The matrix needs to be positive, semi-definite, so we set the non-triad diagonal members to 0.1.

Chord templates:

The chord templates are the theoretical chroma vectors corresponding to the 24 Major and minor triads. A chord template is a 12-dimensional vector which contains the theoretical amplitude values of the notes and their harmonics composing a chord. We consider 24 chord templates corresponding to the 24 Major and minor triads. The amplitude of a note in the template is non-zero if the note belongs to the considered chord (fundamental or harmonic). As in the case of the mean vectors in Method 2, we attribute an amplitude of 0.6^{h-1} to the harmonic h. In Section 4.5, we will compare the system results without considering any harmonic ($nbh = 1$), with 4 harmonics ($nbh = 4$) and with 6 harmonics ($nbh = 6$). In the right part of Figure 4.5, the chord templates for a CM and a Cm chord considering 6 harmonics in the model are represented. The first six harmonics of the notes composing a CM and a Cm chord and their corresponding amplitude are given in Table 4.7. It can be seen that higher harmonics contribute to the pitch class of their fundamental frequencies. For instance, the amplitude of the G is very high in the C major chord (C-E-G) because, besides being a note of the chord, G is a strong harmonic of C. The chord templates for other chords (C#M, ..., BM, C#m, ..., Bm) are obtained from the CM and Cm chords by circular permutation.

Observation probabilities: For each chroma vector, we compute the correlation between the observation vector and each of the 24 chord templates. We obtain 24 values $P(c_i)$, $i \in [1, 24]$, normalized so that $\sum_i P(c_i) = 1$. We now have 24 "pseudo-probabilities" which are used as observation probabilities in the HMM. In what follows, we will call this method "Method 3".

4.4.4 State Transition Probability Distribution

4.4.4.1 Method A: Theoretical Approach Using the Doubly-Nested Circle of Fifths

This method was first proposed by [PC02] for describing the harmonic content of polyphonic music in the symbolic domain. It was then applied in the audio domain in [BP05]. In this approach, the transition probability between two chords is derived from musical knowledge relying on their distance in the doubly-nested circle of fifths (see Figure 4.7).

The doubly-nested circle of fifths depicts relationships among the 12 equal-tempered pitch classes comprising the chromatic scale. The 24 major and minor triads can be represented as points on two overlapping "circles of fifths", one for major triads, the other for minor triads. The more consonant two chords are, the closer on the double circle of fifths. For instance the CM chord (C-E-G) has two notes in common with the Em chord (**E**-**G**-B). It also has two notes in common with the Am chord (A-**C**-**E**). The CM chord is thus placed between the Am and the Em chords on the double circle of fifths.

The closer two triads are on the circle, the higher the corresponding chord transition value is. Following [BP05], we give to the transition CM-CM a probability of 12, CM-Em =

11, and then clockwise in a decreasing manner, until CM-FM# = 0. From this pair of chord, the value of the corresponding chord transition probabilities starts increasing again, starting with CM-Bbm = 1 until CM-Am = 11. These probabilities are normalized so that they sum to unity.

Although we do not know which state is going to follow another, musical rules allow us to make hypotheses that some chord transitions are more probable than others. For instance, especially in popular Western music, an AM chord is more likely to be followed by a F#m or DM chord than by a G#M chord. The corresponding state transition matrix is represented in the left part of Figure 4.8.

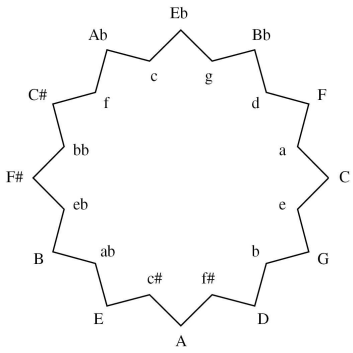

Figure 4.7: Doubly-nested circle of fifths. Adapted from [BP05].

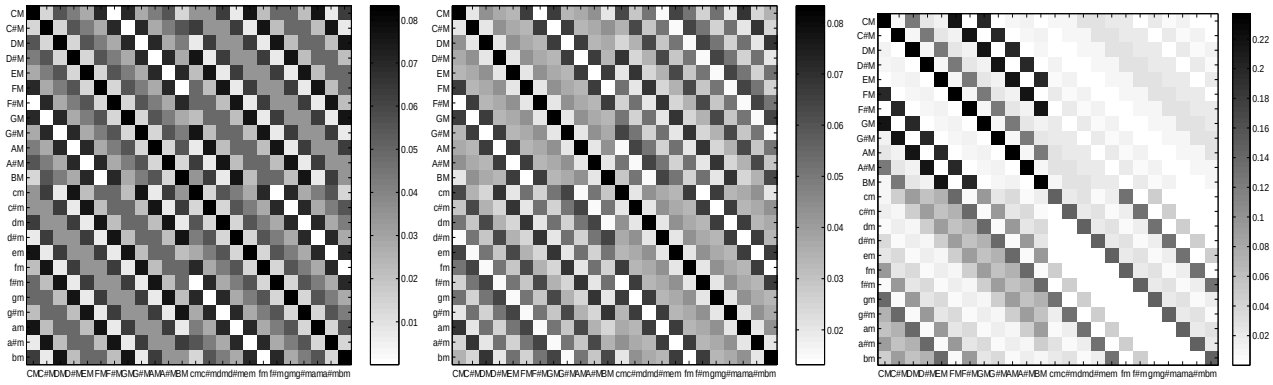

Figure 4.8: State transition matrix between the 12 major and the 12 minor chords. Dark marks indicate high values in the transition matrix. Horizontal axis from left to right and vertical axis from top to bottom: chords (CM, C#M, BM, ..., Cm, ..., Bm). From left to right: method A, method B, and method D.

4.4.4.2 Method B: Cognitive Approach Using Correlation Between Key Profiles

In [Kru90], music-psychologist Krumhansl studies the proximity between the various musical keys using correlations between key profiles obtained from perceptual tests. The probe tone ratings [KK82] represent the stability of each semitone pitch-class relative to a given key (see also Chapter 6, Section 6.2.1.1). These probe-tones ratings are used to obtain a quantitative measure of the distances between keys. Krumhansl & Kessler compute the correlation between profiles for each possible pair of major and minor keys, relying on the idea that two keys are close if they impose a similar pattern of relative stability on the tones. Table 4.8 gives the numerical values corresponding to key profile correlations for CM and Cm keys.

These key profile correlations are used in [NM06] to derive a key transition matrix in the context of local key estimation as described below. In order to have probabilities, all the values are made positive by adding 1, and then normalized to sum to 1 for each key. This results in 24-dimensional vectors that express how likely the music moves from a given key to another at the next time step. The 24-dimensional vectors can be circularly shifted to give the transition probabilities for keys other than CM and Cm. A key transition matrix of size 24 x 24 is built from these 24-dimensional vectors.

In our experiments, we obtained good results for chord estimation using the key transition matrix from [NM06] as a chord transition matrix. This matrix is represented in the middle part of Figure 4.8.

4.4.4.3 Method C: Trained Approach Using the EM Algorithm

This approach uses the transition matrix provided by the training of the HMM using the Expectation Maximization (EM) algorithm, *i.e.* the system is trained using on the one side the succession of chroma vectors extracted from the audio signal and on the other side the corresponding chord labels.

The expectation maximization algorithm [GM99a] is an efficient iterative procedure for finding maximum likelihood estimates of parameters in statistical models, where the model depends on unobserved (hidden) variables. Each iteration of the EM algorithm consists of two processes: the E-step, and the M-step. In the expectation (E)-step, the missing data Q (for us the unknown chord labels) are estimated given the observed data O (the observed chroma vectors) and current estimate of the model parameters θ. In the maximization-(M) step, the parameters are computed by maximizing the expected log-likelihood found in the E-step. Equation (4.1) expresses the complete-data log likelihood as a function of old and new parameters, θ_{old} and θ. At each step the old parameters are fixed and θ is adjusted to maximize $logP(O,Q|\theta)$ in expectation.

$$E[logP(O,Q|\theta)] = \sum_{Q} P(Q|x,\theta_{old}) log(P(O|Q,\theta)P(Q|\theta) \tag{4.1}$$

Convergence is assured since the algorithm is guaranteed to increase the likelihood at each iteration. The specific application of EM to find maximum-likelihood parameter estimates for a hidden Markov model is known as the Baum-Welch, or forward-backward algorithm.

Table 4.8: Krumhansl's correlations between key profiles for CM and Cm keys, from [Kru90].

	C major	C minor
CM	1.000	0.511
C#M	-0.500	-0.158
DM	0.040	-0.402
D#M	-0.105	0.651
EM	-0.185	-0.508
FM	0.591	0.241
F#M	-0.683	-0.369
GM	0.591	0.215
G#M	-0.185	0.536
AM	-0.105	-0.654
A#M	0.040	0.237
BM	-0.500	-0.298
Cm	0.511	1.000
C#m	-0.298	-0.394
Dm	0.237	-0.160
D#m	-0.654	0.055
Em	0.536	-0.003
Fm	0.215	0.339
F#m	-0.369	-0.673
Gm	0.241	0.339
G#m	-0.508	-0.003
Am	0.651	0.055
A#m	-0.402	-0.160
Bm	-0.158	-0.394

4.4.4.4 Method D: Trained Approach Using the Chord Transcription

As opposed to the previous method, this approach is only based on symbolic information, *i.e.* the chord labels transcription of the training set. From the succession of transcribed chord labels over time, we derive an "annotation" transition matrix which is, as in the previous case, specific to the training set (in our case the Beatles corpus). We want to learn from the training set the probabilities of transiting from one chord to another. We achieve this by counting the number of occurrences of each chord transition in the training set. Our goal is to construct a 24-dimensional matrix T that indexes all the chord transitions. However, because the distribution of musical keys is not homogeneous in the training set, we are likely to favor specific chord transitions[9], and therefore the transition matrix will be unbalanced. In order to face this problem, we only consider **relative chord transitions** (GM $\rightarrow$ CM transition is considered as equivalent to CM $\rightarrow$ FM). We denote by $T(i,j)$ the value of the transition matrix that represents the probability of transiting between chord i at time $t-1$ to chord j at time t. The indexes $i,j \in [1,12]$ represent the Major (M) chords, $i,j \in [13,24]$ the minor (m) chords. The matrix is therefore composed by four sub-matrices that represent transitions between M to M, m to m, M to m and m to M chords. These four cases are processed separately.

1. We first select from the training set all chord transitions belonging to a specific case (MM, mm, Mm, mM).
2. For each chord belonging to a given subset, we then compute the relative chord transitions. Each chord transition $i \rightarrow j$ is characterized by the equivalent transition from/to a root-note of C. We denote it by $f(i,j)$.
3. We then form a 12-dimensional vector $\tau(k)$ by counting the number of relative chord transitions $f(i,j) = k$.
4. Using these vectors, we form the $T(1, k \in [1,12])$ (MM), $T(13, k \in [13,24])$ (mm), $T(1, k \in [13,24])$ (Mm), $T(13, k \in [1,12])$ (mM).
5. The diagonal of the sub-matrices (self-transition) is processed in a separate way. We set the diagonal values to $1.1 \max(\tau(k))$.
6. The rest of the sub-matrices are constructed by circular permutation.
7. We finally normalize the matrix T so that the sum of each row is equal to 1.

The resulting matrix trained on the test-set presented in Section 4.5.1 is represented in the right part of Figure 4.8. It is interesting to observe the predominance (high transition values in the matrix) of typical transitions in the matrix, such as the II/V/I (transition between Dm, GM and CM) that seems usual in this set of Beatles albums. However, the amount of transitions between Major and minor chords is much lower than the amount of transitions between two Major chords in this training set. It can be noticed, for instance, that the typical transition CM-Am, that frequently arises in songs in the C major key,

[9] For instance, if 90% of the training set is in C Major we are more likely to observe a II/V/I transition in C Major, *i.e.* Dm/GM/CM, than a II/V/I transition in F#M, *i.e.* G#m/D#M/F#M.

are not enhanced in this trained transition matrix. The consequence of that, is a lower estimation rate for tracks with Major to minor chords.

4.4.5 Chord Progression Detection Over Time

In all cases (Method 1, 2, 3, A, B, C or D), the optimal succession of chords over time is found using the Viterbi decoding algorithm [Rab89] which gives us the most likely path trough the HMM states given our sequence of chroma observations.

4.5 Evaluation and Results

This study was initially published in [PP07]. It was the first large-scale evaluation of chord estimation algorithms.

4.5.1 Test Set and Protocol

The system has been tested on a set of 110 hand-labeled files from the first eight albums of the hand-labeled *Beatles test-set* presented in Chapter 2, Section 2.3.2. The chord label accuracy is measured using the measure detailed in Section 2.4.2 of Chapter 2.

4.5.2 Results

The chord estimation results obtained with the various methods are indicated in Table 4.9 and illustrated in Figure 4.9. Note that we present here earlier results published in [PP07]. They were obtained using a FFT-based chroma representation (and not using a CQT-based chroma representation) and correspond to a frame-by-frame analysis (not a beat-synchronous analysis). The purpose here is to compare the various proposed configurations of the HMM[10].

In Table 4.9, we compare the various methods according to the nature of the observation distribution and to the number of harmonics (nbh)considered:

• (Method 1) Gaussian observation distribution with training. For this method, the evaluation has been performed using a 8-folds cross-validation (each album was evaluated using the seven remaining albums as training data).

• (Method 2, $nbh = 1$) Gaussian observation distribution with music theory as proposed in [BP05].

• (Method 2, $nbh = 4$) Our proposal: Gaussian observation distribution with music theory considering the presence of four higher harmonics.

[10]To introduce chord dependency to the meter, we have later used beat-synchronous chroma features. In the next chapter, we will present more recent chord estimation results using beat-synchronous chroma features.

• (Method 3, $nbh = 1, 4, 6$) Our proposal: Observation distribution from correlation with templates combined with music theory considering the presence of one, four or six higher harmonics.

Note that we only present here the results obtained using method B for the transition matrix (see explanations Section 4.5.3.2).

Table 4.9: Chord estimation rate (mean and standard deviation) using methods 1, 2 and 3 for the observation distribution and transition matrix B (theoretical transition matrix based on correlation between key profiles). Rex: exact chord estimation rate. Rct: chord estimation rate including close triads. nbh: number of harmonics considered in the model.

	Method1	Method2		Method3		
		nbh = 1	nbh = 4	nbh = 1	nbh = 4	nbh = 6
Rex	69.95 ± 14.90	61.57 ± 14.72	69.28 ± 11.42	67.54 ± 13.54	70.22 ± 17.01	70.96 ± 19.23
Rct	84.08 ± 9.87	74.67 ± 10.47	81.82 ± 9.91	81.22 ± 9.64	82.57 ± 10.49	86.18 ± 8.67

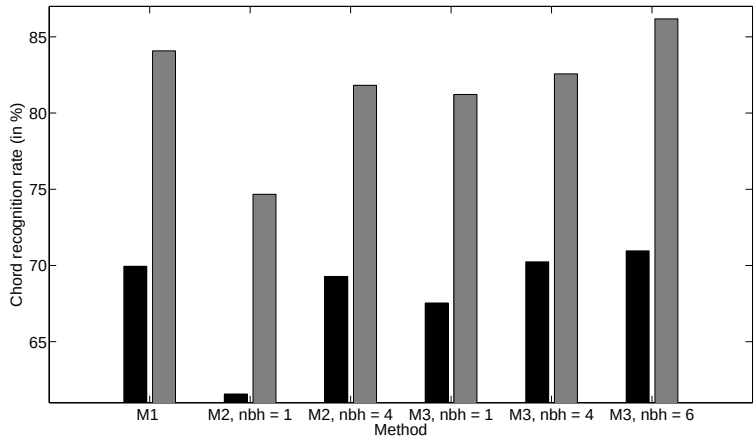

Figure 4.9: Histogram of chord estimation results obtained using the transition matrix based on correlation between key profiles (method B) according to the various methods. From left to right: method 1 (M1), method 2 considering 1 and 4 harmonics (M2, nbh = 1; M2, nbh = 4) and method 3 considering 1, 4 and 6 harmonics (M3, nbh = 1; M3, nbh = 4; M3, nbh = 6). In black: exact chord estimation rate. In grey: chord estimation rate including close triads.

Table 4.10: Statistical Significance (Stat. Sig.) of the difference between the results obtained with several pairs of methods. nbh: number of harmonics considered in the model.

Compared Methods	Stat. Sig.
Method1 - Method2, nbh = 1	yes
Method1 - Method2, nbh = 4	no
Method1 - Method3, nbh = 6	yes
Method2, nbh = 4 - Method3, nbh = 6	yes
Method3, nbh = 4 - Method3, nbh = 6	yes

4.5.3 Analysis of Results

4.5.3.1 Chord Estimation Method

The results obtained with the various methods are pretty close to each other. However, we performed paired-sample t-tests at the 5% significance level and we found that the difference between the results is statistically significant in most of the cases (see Table 4.10).

In our experiments, the best results were obtained with Method 3 (70.96%). Note that there was no training of the observation distribution in this case. Despite the fact that Method 1 uses training (and is therefore likely to fit very well to the characteristics of the Beatles), Method 2 with $nbh = 4$ (which does not use training at all) gives very close results[11]. Note that the difference between the two methods is not statistically significant.

4.5.3.2 Transition Matrix

The best results were obtained using the theoretical transition matrix based on correlation between key profiles (Method B). The transition matrix based on the doubly-nested circle of fifths (Method A) gives slightly lower results. We do not present the results obtained with the two trained matrices (Methods C and D). Although method C is the usual approach and the one used for example in [SE03] [BP05], it did not provide satisfactory results in our evaluation. Method D did not perform well because, as explained in Section 4.4.4.4 some typical transitions are not enhanced in this trained transition matrix. In Chapter 6, we will show how this training method can be improved by taking into account information related to the musical key.

4.5.3.3 Number of Harmonics

Considering the presence of higher harmonics in the model clearly improves the results. For instance, for Method 3, considering 6 harmonics in the templates brings about 5% relative improvement to Method 3 with $nbh = 1$. Note that the difference in the results considering 4 and 6 harmonics, although small, is statistically significant (see Table 4.10). This is even clearer in the case of Method 2 where considering harmonics in the parameters of the model brings about 12.5% relative improvement (compared to Method 2 with $nbh = 1$).

[11] It should be noted however that, although Method 1 and $nbh = 1$ is very close to the one presented in [BP05], we did not recover the high results reported in [BP05].

4.5.4 Discussion

4.5.4.1 Chord Confusions Due to Ambiguous Mapping

As it can be seen in Table 4.9, the standard deviation of the results is relatively high (up to 19%) independently from the chosen method. A deeper analysis of the results shows that the errors come from a subset of songs which possess specific characteristics described below.

Concerning partial chords, we obtain for instance less than 3% of chords correctly identified on the song *Love You To* from the Beatles album *Revolver*. Provided annotation indicates that almost all the chords of this song but a few are Cmin(*b3) chords, *i.e.* a triad without the third note (C-G). In such a case, it is difficult to make a decision between major and minor chords in the absence of musical key information. For this song, our system in fact recognized in all cases a CM chord instead of a Cm chord, resulting in a low estimation rate.

As mentioned earlier, because of our limited chord dictionary, a mapping was performed between complex chords and their root triad. The chord type distribution in the test-set is unbalanced and, even if the majority of the songs in the evaluation test-set are composed of triad chords, some of them contain many partial or complex (non-triads) chords. The system sometimes recognizes other triads than the root triad of the complex chord analyzed, which decreases the estimation rate. For instance, the Beatles song *Ask Me Why* contains many G#min7 chords (G#-B-D#-F#). This complex chord comprises a G#m chord (G#-B-D#) and a BM chord (B-D#-F#). The theoretically correct answer depends on the tonal function of the chord in the harmonic progression. Modeling chord sequences using longer dependencies between chords, using for instance probabilistic N-grams, would help characterize the complexities of harmonic progressions in Western tonal music.

4.5.4.2 Neighboring Triad Confusions

It can be noticed that most of the chord errors correspond to harmonically close triad confusions:

- Parallel Major/ minor chords (EM being confused with Em),
- Relative chords (Am being confused with CM),
- Dominant chords (CM being confused with GM),
- Subdominant chords (CM being confused with FM).

If the system does not recognize exactly a chord but makes such confusions, the result can still be useful for higher-level structural analysis such as key estimation, harmony progression or segmentation. Table 4.9 shows that if we consider close triads estimation as correct, the estimation rate of method 3 reaches up to 86%. It also becomes now the

method with the smallest standard deviation, 9%. This point will be further discussed in the next chapter.

4.5.4.3 Passing or Missing Tones

In the Beatles song *Till There Was You*, there is a repeating pattern beginning by an FM chord that has a duration of two beats. The system estimates the following chords: FM-Dm. If we listen to the music, we can hear that on the first two beats, the guitar is playing a broken FM chord (F-A-C). On the second beat, the C note is not present anymore. A musician would naturally label the two chords as a FM chord, ignoring the fact that there are missing notes (because it is the same harmony). However, the signal features only take into account notes that are present in the signal. As a result, the estimated chords do not match exactly those of the ground truth. Conversely, non-chord tones such as neighbor tones, anticipation, passing tones, suspension and escape tones that occur in the melody and do not belong to the harmony may also confuse the harmony. This example leads to the relevant question of how to evaluate the performances of a chord estimation system. The ground truth is provided by trained musicians who not only take into account the notes present in the signal but also the harmonic context to label the chords, ignoring the addition or the deletion of some notes in their annotation. This complicates the evaluation of the algorithm.

4.5.4.4 Limitation of the Chroma-Based Approach for Inharmonic Sounds

It is interesting to notice that we obtain much better results for the five first Beatles albums than for the others (from the *Norwegian Wood (This Bird Has Flown)* on 1965's *Rubber Soul* album). The reason for this may come from the extended use of the Indian sitar instrument[12] and various percussive instruments such as bells, wood blocks or congas that cause transients. Since the chroma-based approach strongly relies on the presence of harmonic sounds, the use of chroma-based signal features would ideally require a pre-processing step that effectively reduces transients and noise. We plan to concentrate on this point in future work.

4.6 Conclusion

In this chapter we have proposed and compared several methods for the automatic estimation of chord progression of an audio signal of music. All the methods are based on a chroma representation of the audio signal and on modeling of the sequence of observation using a hidden Markov model. The methods have been compared through a large-scale

[12]The sitar is a stringed instrument that uses sympathetic strings in addition to regular strings. This produces a very lush sound with complex, competing harmonic components.

evaluation. We have presented here the results that were originally published in [PP07]. To our knowledge, it was the first attempt to evaluate chord estimation algorithms on such a large test-set. The best results are obtained with the modeling of the observation probabilities using a normalized correlation with a set of extended chord templates and a cognitive-based transition matrix. The templates are extended by considering the presence of higher harmonics of each pitch note of a chord. The transition matrix is derived from cognitive experiments on the perception of musical key.

In our experiments, we have found that music knowledge-based parameters work at least as well as trained parameters. However, we believe that the training could still be exploited and yield to higher results. The best results for the chord estimation task obtained in the Music Information Retrieval Evaluation eXchange (MIREX) contests were obtained by trained systems. Moreover, the proposed music knowledge-based parameters can only be used for a chord lexicon reduced to the 24 major and minor triads.

However, since we only consider these 24 triads, we will use in the rest of the present work the HMM-based approach relying on chord templates since it gives satisfactory results without requiring any training data. We will use the transition matrix based on Krumhansl's key profiles because we believe that this matrix, as well as the one based on the circle of fifths, characterizes well harmonic relationships in a large part of Western tonal music styles including classical and popular music, without requiring any training data. This will allow us to work on other styles of music than popular music (see Chapter 6). It is important to note that the approaches that will be presented in the next chapters can be extended to a larger chord lexicon and do not depend on the choice of the chord estimation method or the choice of the chord transition matrix.

A limitation of the model comes from the confusion between the various interpretations one can make about chords. A solution would be to integrate extra (context) information such as musical key information. The integration of metrical information could also increase the robustness of the system. This is the points we will focus on in the next chapters.

Chapter 5

Joint Estimation of Chords and Downbeats

In this chapter, we present a new technique for joint estimation of the chord progression and the downbeats from an audio file. Musical signals are highly structured in terms of harmony and rhythm. In this chapter, we intend to show that integrating knowledge of mutual dependencies between chords and downbeats allows us to improve the estimation of these musical attributes. For this, we propose a specific topology of hidden Markov models that enables modeling chord dependency on the metrical structure. This model allows considering pieces with complex metrical structures such as beat insertion, beat deletion or changes in the meter. It is evaluated on a large set of popular music songs from the Beatles that present various metrical structures. We compare a semi-automatic model in which the beat positions are annotated with a fully automatic model in which a beat tracker is used as a front-end of the system. The results show that the downbeat positions of a music piece can be estimated in terms of its harmonic structure and that, conversely, the chord progression estimation benefits from considering the interaction between the metric and the harmonic structures.

Contents

5.1 Introduction

The previous chapter has been devoted to chord progression estimation. In this dissertation, we are interested in understanding how various musical attributes may interact with each other. In this chapter, we focus on the problem of estimating simultaneously two musical attributes: the chord progression, which is related to the harmony, and the downbeats, which are related to the metrical structure. A piece of music can be characterized by its chord progression that determines the harmonic structure. The chord progression is closely related to the metrical structure of the piece [Got01]. For example, chords will change more often on strong beats than on other beat positions in the measure. Most of the previous studies deal with various musical attributes independently. However, harmony and meter are deeply related to each other and their automatic estimation should be improved by exploiting their interrelationship. In this chapter, we present a system that allows the simultaneous estimation of the chord progression and the downbeats from an audio file. Most of the previous works on downbeat detection have dealt with constant meter pieces. A contribution of this chapter is that we consider the problem of complex meter (*e.g.* changes in the meter, insertion or deletion of beats). We also consider the problem of imperfect beat tracking. The model is evaluated on a large set of popular music songs and gives very interesting results on pieces with complex metrical structure. This chapter is based on publications [PP08b] and [PP10].

The major contributions of this chapter are the following:

1. We provide a detailed review of the previous works related to the problem of downbeat estimation, including interaction between harmony and meter.

2. We present an approach to the chord progression and the downbeat tracking estimation problems, which are jointly considered using a specific topology of hidden Markov models.

3. The proposed model can be used for pieces containing changes in the meter.

4. The system can handle real situations, when using an imperfect beat tracking as a front-end of the system.

5. We have annotated the beats and the downbeats of a large set of popular music songs.

6. This allows us to provide a quantitative evaluation of our model considering various cases of meter.

7. We provide a deep analysis of chords/downbeats interaction results.

8. We compare the newly proposed model with the state-of-the-art and show that it presents improvements.

9. We provide a discussion of the proposed model.

Organization of the chapter:

This chapter is organized as follows. First, in Section 5.2, we provide a review of previous works related to the problem of downbeat tracking. We then introduce in Section 5.4 a probabilistic model for simultaneous chord progression and downbeat position estimation. This model encodes contextual information in the state transition matrix; this is detailed in Section 5.4.5. In Section 6.3.3, we present our approach to estimate the two considered musical attributes (chords and downbeats) using the Viterbi decoding algorithm. In Section 5.6, the proposed model is evaluated on a set of hand-annotated songs from the Beatles. A conclusion that underlines the advantages and the limits of the proposed model closes the chapter.

5.2 Related Work

The problem of tracking beat and tempo in audio signals is addressed in a large number of previous works. Even if it has drawn less attention than beat tracking, downbeat detection is an interesting problem that deserves to be carefully studied and a number of contributions dealing with various aspects of this problem have already been proposed. This is not surprising since downbeat positions knowledge may be useful in various applications within the context of music information retrieval. It may facilitate fully automated rhythmic pattern analysis, as in the work of Ellis & Arroyo [EA04] where a representation of the drum patterns is used as a space for genre classification. It can be used for automated rhythmic transformation of musical audio, as in the work of Hockman & Bello [HB08] where a technique for automatic mixing and synchronization between two musical signals in a disc jockey application is presented. It may serve to partition the signal into segments of lengths that have a musical meaning in structural audio segmentation, as in the work of Bartsch & Wakefield [BW05]. It may also be used in intelligent computer accompaniment, as in the work of Goto [Got01] where the downbeats are used to produce an intelligent drum machine that can play drum patterns in time to input musical audio signals without drum-sounds.

The downbeat tracking problem has first drawn the attention of researchers working with MIDI format. For instance Temperley & Sleator [TS99] propose a computational system for analyzing metrical and harmonic structure. The program takes as input a symbolic representation of music. The metrical structure produced by the algorithm consists of several levels of beats, including the downbeat positions. The approach is based on preference rules. The inference of the metrical structure relies on three rules: the *event rule*, the *length rule*, and the *regularity rule*. For a given piece, all possible analyses are considered. The analysis that best satisfies the rules is selected among the others. The performances of the model are illustrated on some examples.

In this dissertation, we are interested in working directly on the audio signal. The first downbeat tracking system that works reasonably well on audio was presented by Goto & Muraoka in [GM99b]. In this work, a complex agent-based model for detecting a hierarchical beat structure in musical audio signals without drum-sounds is proposed.

The system tracks beat structure at the quarter-note, the half-note and the measure levels, and operates in real-time. The analysis is restricted to pieces having a 4/4 time-signature and the tempo is assumed to be roughly constant within the range of 60 to 120 beats per minute (bpm). The hierarchical beat structure is identified relying on musical knowledge. The system is based on an architecture where multiple agents track alternative meter hypotheses. The provisional beat times are a hypothesis at the quarter-note level and are inferred by an analysis of onset times. Short-term spectral frames are peak-picked and then "histogrammed" into beat length segments, where chord changes are used to infer higher level metrical structure. In the same way that untrained music listeners, who cannot identify chord names but are able to perceive harmony and chord changes, the chord changes detection method does not require chord names to be identified. The approach is tested on 40 popular music songs and estimates correctly the downbeat positions for 94.1% of the songs for which the quarter-note level and the half-note level have been correctly estimated. The experiments show that both chord-change possibilities based on the eighth-note-level knowledge and on the quarter-note-level knowledge are necessary for determining the hierarchical beat structure. The method was further combined with a previous beat-tracking system designed to process real-world audio signals with drums [GM94] [GM96]. Musical knowledge of chord changes and musical knowledge of drum patterns are selectively applied according to the presence or absence of drum-sounds. This results in a single system that can recognize the hierarchical beat structure of music with drums and music without drums by using three kinds of musical knowledge: onset times, chord changes and drum patterns.

Previous works that specifically address the problem of downbeat tracking can be found in the literature. Most of them rely on prior knowledge such as tempo, time-signature of the piece or hand-annotated beat positions. Allan [All04] presents a model that uses an autocorrelation technique to determine the downbeats in musical audio signals for which beat positions are known. The system relies on the assumption that a piece of music will contain repeated spectrally similar patterns. The boundaries of those patterns are assumed to fall on metrical hierarchical boundaries (bars). Bar boundaries are identified from the known beat positions by measuring the Euclidean distance between grouped beat length spectral segments of varying lengths at incremental offsets. It has been tested on 42 different pieces of music at various metrical levels, in several genres. It achieves a success rate of 81% for pieces in 4/4 time-signature and needs more testing on ternary time-signatures.

Hand-annotated beat positions are not needed in the model proposed by Jehan in [Jeh05]. This work proposes an unbiased and predictive approach for downbeat tracking that combines psychoacoustic models of music listening with time-lag embedded segment learning. The model is tempo independent and does not require beat tracking. It however requires some fair amount of prior knowledge acquired through listening or learning during a supervised training stage where downbeats are hand-labeled. The model has only been applied to music in 4/4 meter. To demonstrate its performances, it is applied to two complex musical cases for which the downbeat cannot be interpreted through harmonic shift or a generic "template" pattern: a song characterized by its repetitive single chord and syncopated rhythmic pattern and a rhythmically complex piece of Brazilian music. However, the model is not quantitatively evaluated.

A recent method that segments the audio according to the position of the bar lines (downbeats) has been presented by Gainza *et al.* in [GBC07]. The model does not depend on the presence of percussive instruments and allows moderate tempo deviations. The downbeat detection is based on three independent tasks: bar line detection, anacrusis detection and bar line alignment. The bar length and the anacrusis beats are identified using an audio similarity matrix. The bar length is determined by computing the length of the most repeating segment within a range of bar length candidates, which are derived from tempo and time signature ranges. A vector of anacrusis candidates is generated, on which an anacrusis detection function is applied. The position of each bar line is then predicted by using on the one hand prior information about the position of previous bar lines, and on the other hand the estimated bar length. Finally, each bar line is estimated by aligning the predicted bar line position to the most prominent value in an onset detection function within a window centered at the predicted bar line. The approach is evaluated on a set of 9 popular music excerpts from which the downbeats have been manually annotated. It shows that the detection of the bar length is accurate but the detection of the anacrusis is not. The model has the advantage that it does not require tempo estimation and that the alignment of the bars allows moderate tempo deviation. However, it may be badly affected by time signature or abrupt tempo changes.

Contrary to some previous mentioned methods such as [GM99b] [Jeh05], Klapuri *et al.* [KEA06] propose an approach that allows tempo deviation and is not restricted to a particular time signature (typically 4/4 in the previous approaches). This work is not restricted to downbeat tracking but analyzes the musical meter into three different metrical levels: tatum, tactus and measure level. A probabilistic model that encodes prior musical knowledge jointly estimates the period-length and phase of each level, by taking into account the temporal dependencies between successive estimates. The downbeats are identified by matching rhythmic pattern templates to a mid-level representation. The proposed downbeat tracking approach is evaluated on a manually annotated database of 320 one-minute long excerpts of musical signals from various genres. It is noticed that pitch analysis should be used to estimate more accurately the downbeats.

Ellis & Arroyo [EA04] also use a "template-based" approach in a drum-pattern classification and generation task. For this a collection of drum patterns is created. The downbeat of an input drum pattern is defined as the beginning of a looping drum pattern. To estimate this point, the input pattern is cross-correlated with reference patterns. The method is evaluated on a corpus of 100 drum tracks from real pieces of different genres, encoded as MIDI files. The algorithm estimates correctly the downbeat positions of half of the tracks for which the tempo and the pattern length have been correctly estimated. It is concluded that the downbeat detection would require a more sophisticated approach such as training.

The above-mentioned rhythmic pattern approach [KEA06] is compared with an approach based on a spectral difference between band-limited beat-synchronous analysis frames proposed by Davies & Plumbley in [DP06]. The sequence of beat positions of the input signal is required and the time-signature is to be known *a priori*. The input signal is partitioned into band-limited beat length frames. Relying on the musical knowledge that lower frequency bands are perceptually more important, information within the range

$0 - 1.4$kHz is preserved. The Kullback-Leibler divergence between successive beat frames is computed in order to form a spectral difference function. The beats that globally lead to most spectral change are selected as downbeats. The model is evaluated against the one presented in [KEA06] on a subset of the database originally presented in [Hai04], that consists of 181 excerpts of six musical genres (rock, dance, jazz, folk, classical and choral). It obtains an overall accuracy of 53% rising to 81% for cases where beat tracking is accurate, comparing favorably with the state-of-the-art [KEA06], which obtains respectively 40.8% and 69.9%. This downbeat extraction method is employed in [HB08] for the purpose of automatic mixing and synchronization between two musical signals. We consider this approach to be the state-of-the art. One of the drawbacks of this approach is that any omitted beat or change in tempo or time-signature causes errors from which the model cannot recover. Moreover, it is limited to cases where the time-signature does not change and the tempo is approximately constant.

The strong relationship between the chord progression and the metrical structure of a musical piece has already been explored in previous works [TS99], [Mad06], [SW05], [PEBB05] or [RS03].

In the work of Temperley & Sleator [TS99], information about the metrical structure is used during the analysis of the harmonic structure: the "strong-beat rule" stipends that it is preferable to start chord spans on strong beat scores. In this model, there is no complete interaction between the harmonic and the metrical information. Indeed, if harmonic analysis uses metrical information, the metrical analysis process does not use harmonic information. This is viewed as a drawback of the model.

Drawing on the prior idea of Goto [Got01], Shenoy & Wang [SW05] present a framework that provides the hierarchical rhythm structure representation of a piece of music at the quarter-note, the half-note and the measure levels. The aim of this work is to determine the key, chords and hierarchical rhythm structure of a music signal by combining low-level features with high-level music knowledge in a rule-based approach. Harmonic and metric information are estimated in a mutually informing manner. A first estimation of the chord progression is provided using beat-length chroma features. The measures boundaries are then estimated relying on the music knowledge that chords are more likely to change at the beginning of a measure than at other beat positions [Got01]. Assuming a 4/4 time-signature, all possible patterns of boundary locations that have integer relationships in multiples of four are computed. The pattern with the highest count is selected as the one corresponding to the pattern of actual measure boundaries. Finally, the measures boundaries are used to correct possible chord errors relying on the rule that chord changes are more likely to change at the beginning of the measures than other positions of half-note times. The system works reasonably well on popular music assuming a constant 4/4 meter and a fixed tempo constrained between 40 and 185 bpm. Tests on a set of 30 popular English songs lead to an accuracy of 93% for the downbeat tracking. However, it is noted that the model cannot be used to analyze music more rhythmically and tonally complex. Moreover, possible beat detection errors are systematically propagated into the downbeat tracking stage.

This is a typical drawback of rule-based approaches. One of the main drawbacks of rule-based approaches is that errors are irreversibly propagated from one step to another.

Statistical approaches, including graphical and Bayesian models, are more flexible than rule-based approaches and offer large opportunities to explore the interaction between low-level features with high-level music information. An example of such a work related to the issues addressed in this chapter is the one presented by Paiement *et al.* in [PEBB05]. It considers the interaction between the harmonic and the metrical structures using a graphical (probabilistic) model where contextual information is used to model the chord progression. It is related to our work in the sense that information related to the meter is used for modeling the chord progressions. However, the approach is different. It is not based on a HMM but the strong relationship between the chord progression and the meter of the piece is embedded in a tree structure that captures the chord structure in a given musical style. The main assumption behind the model is that conditional dependencies between chords in a typical chord progression are strongly tied to the metrical structure associated to it. In this model, a chord progression is seen as a two-dimensional architecture. Each chord in the chord progression depends both on its position in the chord structure (global dependencies) and on the surrounding chords (local dependencies).

The various presented methods for downbeat estimation from audio files are summarized in Table 5.1.

Table 5.1: Summary of downbeat estimation methods.

Reference	Method	Meter	Knowledge applied	Evaluation Material
Goto & Muraoka [GM99b]	agent-based model	constant 4/4	musical knowledge of chord changes and musical knowledge of drum patterns	40 popular music songs
Allan [All04]	autocorrelation technique	constant 4/4 and 3/4	Euclidian distance between grouped beat length spectral segments	42 different pieces in several genres
Ellis & Arroyo [EA04]	template-based approach	finding the beginning of a looping drum pattern	cross-correlation with reference drum patterns	100 drum tracks from real pieces of different genres, encoded as MIDI files
Jehan [Jeh05]	unbiased and predictive approach	constant 4/4	prior knowledge acquired through listening or learning	two complex musical songs
Shenoy & Wang [SW05]	rule based	constant 4/4	musical knowledge of chord changes	30 popular English songs
Klapuri *et al.* [KEA06]	probabilistic model	no restriction	joint analysis at three different time scales, encode musical knowledge	320 one-minute long excerpts from various genres
Davies & Plumbley [DP06]	spectral difference between beat synchronous analysis frames	constant 3/4 or 4/4	musical knowledge that lower frequency bands are perceptually more important	181 files from [Hai04] database
Gainza *et al.* [GBC07]	similarity matrix	constant	exploit the self-similarity nature of the structure of music	9 popular music excerpts
Papadopoulos & Peeters [PP08b]	double-state HMM	constant 4/4	simultaneous estimation chords/downbeats	66 Beatles songs
Papadopoulos & Peeters [PP10]	double-state HMM	no restriction, variable	simultaneous estimation chords/downbeats	169 Beatles songs

5.3 Proposed Approach

The review of a number of previous works on downbeat estimation shows that the downbeat positions are deeply related to the chord progression of a piece of music. Our purpose is to show how information related to the metric and the harmony of a piece of music interact and how this can be used in a mutually informing manner to improve both the estimation of the chord progression and the downbeat positions.

There are several issues related to this problem that still need to be addressed. In particular, most of the previous works assume constant tempo and/or time signature. Any omitted beat or change in tempo or time-signature causes errors from the downbeat extraction model cannot recover.

We describe a model that addresses these issues. In our model, the time signature is not found at the beginning of the piece and then extrapolated metronomically through the entire piece, but the downbeat positions are adjusted to the music. Our model allows considering pieces with complex metrical structures including changes in the meter from 3/4 to 4/4 time-signature but also exceptional situations such as for instance the insertions of a measure in 1/4 in a 4/4 meter passage. Figure 5.1 illustrates the various metrical structure cases that we consider in our model:

- a) constant 4/4 meter
- b) constant 3/4 meter
- c) variable 4/4 meter with passages in 3/4 meter
- d) variable 3/4 meter with passages in 4/4 meter
- insertion of one measure in a different time-signature in a constant 3/4 or 4/4 meter passage. This can be viewed as :
  - e) either 1 or 2 inserted beats within a constant meter passage
  - f) or 1 or 2 deleted beats within a constant meter passage.

Our model can also handle with errors in the beat tracking stage such as beat insertion or beat deletion due in general to tempo deviation, *e.g.* music tempo speed up or slow down that is not detected by the beat tracker (see Figure 5.2).

In the previous chapter, we have seen that the hidden Markov models have often been used to model the chord progression of an audio file (see for example [SE03], [BP05], [Lee08]). One of the reasons why the chord progression is modeled by an HMM is that the observation of a given chord depends on the previous chord according to musical composition rules which can be modeled in a transition matrix. In this chapter, we extend the previous method for chord estimation and propose a specific topology of HMM that allows us to extract simultaneously the chord progression and the downbeats from an audio file. For this, we first extract a set of feature vectors that describe the signal using a method presented in Chapter 3 of this dissertation. The

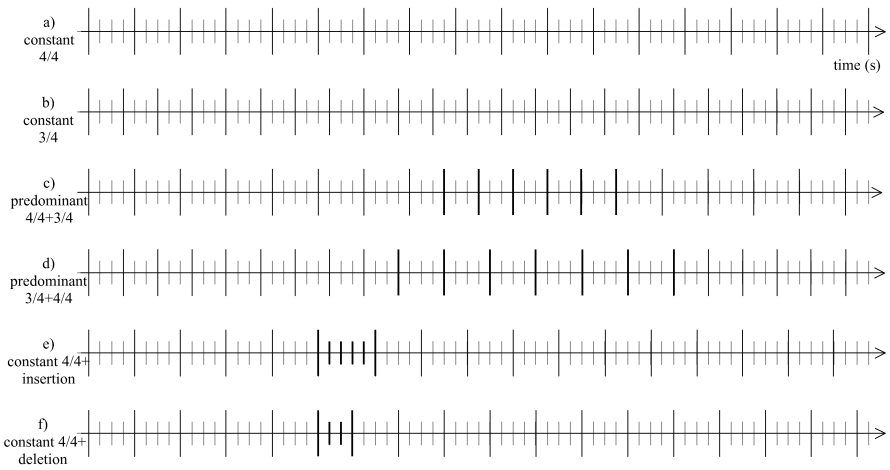

Figure 5.1: Various cases of metrical structure considered in our model.

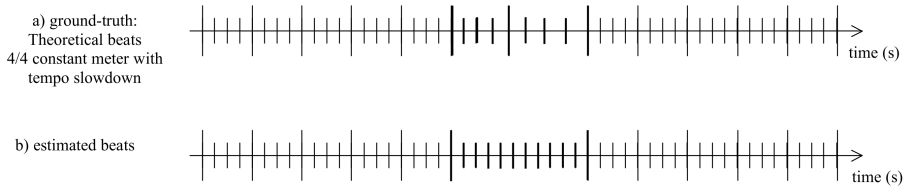

Figure 5.2: Example of beat insertion in the estimated beats due to slow down of the music tempo that is not detected by the beat tracker.

chroma vectors are averaged according to the tactus/tatum positions that have been extracted using the method proposed in [Pee] and [Pee09]. The chord progression is represented using a hidden Markov model that takes into account global dependence on meter. We present a "double-states" HMM where a state is a combination of a chord type and a position of the chord in the measure. Harmonic and metrical structure information are encoded in the transition matrix. In order to take several cases of metrical structure into account, two transition matrices are proposed. Using a Viterbi decoding algorithm, the most appropriate matrix is selected. We then obtain simultaneously the most likely chord sequence and downbeat positions path over time. The flowchart of the system is represented in Figure 5.3.

5.4 Model

5.4.1 Extraction of Beat-Synchronous Chroma Features

The front-end of our system is based on the extraction of a set of feature vectors (*chroma vectors*) that represent the audio signal. We refer the reader to Chapter 3 for more details about chroma features extraction. The results presented at the end of this chapter have

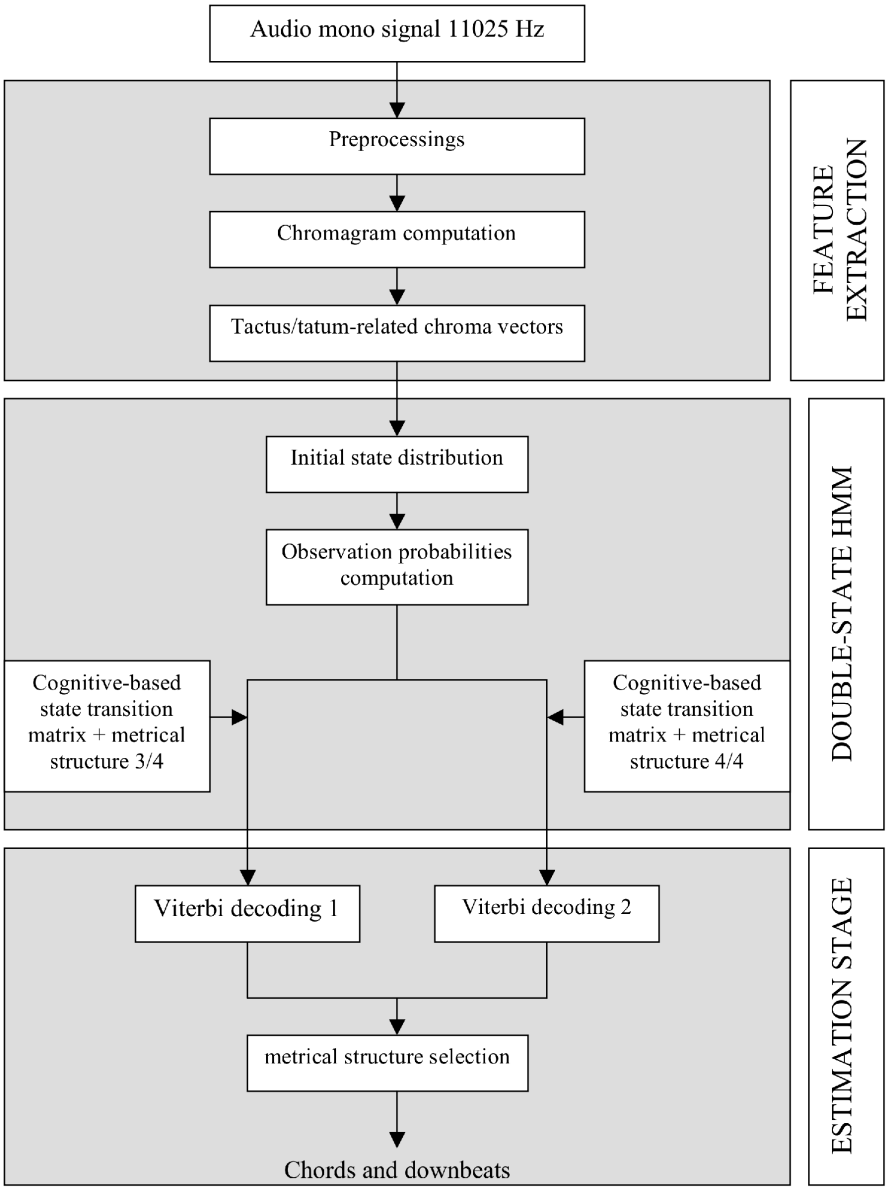

Figure 5.3: General flowchart of the proposed model for simultaneous chord progression and downbeat estimation.

been obtained using beat synchronous, 12-dimensional constant-Q-based chroma features.

Many approaches related to chord estimation have used beat-synchronous chroma features, as for instance [BP05], [RSN08], [SIY+08], [YKK+04], [ZR07] for chord estimation, [Bel07], [Ell07], [EP07], [ECM08] for music similarity and cover song identification or [Mad06], [MKL06] [SW05], [BW01], [BW05] for music segmentation and music structure detection.

As seen in chapter 3, [BP05] argue that beat-synchronous analysis frames help over-

come noise introduced by transients components in the sound (drums and guitar strumming) and short ornamentations, thus minimizing the effect of local variations. Averaging the analysis windows between two beats results in some smoothing. In music similarity applications, the use of a beat-synchronous representation enables overcoming variability in tempo between two music pieces [Ell07].

In our case, we want to study the relationship between the chords and the metrical structure. We thus need some observation features that are related to the meter. The frame by frame analysis does not fit our needs: we need to proceed to a beat synchronous analysis. To this end, the chromagram is averaged so that we obtain one feature per tactus/tatum[1]. In the present work, we use the beat tracker proposed in [Pee] and [Pee09] as a front end of the system. Briefly, [Pee] proposes a method that aims at detecting tempo at the tactus level for percussive and non-percussive audio. The front-end of the system is based on a proposed reassigned spectral energy flux for the detection of musical events. The dominant periodicities of this flux are estimated by a combination of discrete Fourier transform and frequency-mapped autocorrelation function. The most likely meter, beat and tatum positions over time are then estimated jointly using meter/beat subdivision templates and a Viterbi decoding algorithm. The beat tracking is then performed using a method adapted from a P-SOLA glottal closure instant detection using estimated tempo and local maxima of the onset-energy function.

For each tactus/tatum position p_k of the piece, we compute a chroma vector C_k. Each bin $C_k(l), l = [1;12]$ is obtained by computing the average of the N_k chroma vector bins $C_n(l)$ over the considered tactus position and the following one:

$$C_k(l) = \frac{1}{N_k} \sum_{p_k \leq n < p_{k+1}} C_n(l) \tag{5.1}$$

The feature extraction stage is represented in Figure 5.4.

In our study, we have considered two cases. The chromagram has been averaged with respect to the beats or quarter notes (*tactus*) in the first case, and with respect to the eighth notes (*tatum*) in the second case.

5.4.2 Overview of the Model

The proposed model for simultaneous estimation of chords and downbeats is an extension of the chord estimation model proposed in the previous chapter. We consider an ergodic $I * K$-states HMM where each state s_{ik} is defined as an occurrence of a chord c_i, $i \in [1;I]$ occurring at a "position in the measure" (position of a beat or tatum inside a measure) pim_k, $k \in [1;K]$:

$$s_{ik} = [c_i, pim_k].$$

In our experiments, our chord lexicon is still composed of $I = 24$ Major and minor triads (C Major, ..., B Major, C minor, ..., B minor). We keep the notations introduced in the previous chapter: CM, ..., BM for major chords, Cm, ..., Bm for minor chords.

[1]The tactus/tatum positions are either considered as input to the system in the case of semi-automatic analysis or obtained using a beat tracker as a front-end of the system in the case of fully-automatic analysis.

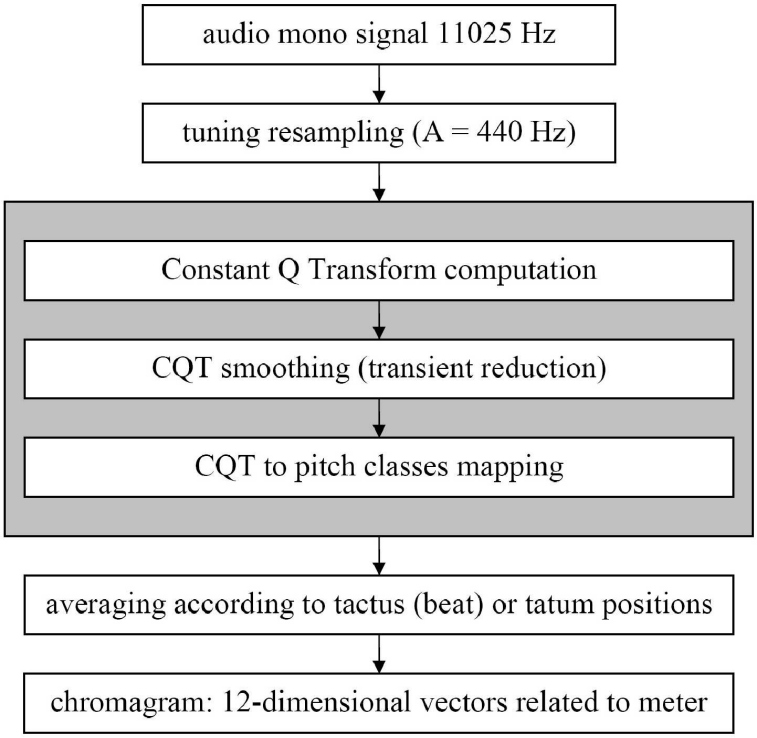

Figure 5.4: Chroma features extraction.

In the proposed model, chord changes can only occur on beats or half beats, which corresponds respectively to the tactus and the tatum positions in the test set. In the rest of the text, the positions in the measure where chord changes occur will be referred to as "position in the measure" and denoted by *pim*. We consider here pieces predominantly in 3/4 or predominantly in 4/4 meters. In both cases, the transition matrix will allow 4 beat positions in the measure. $K = 4$ if we consider the tactus-level and $K = 8$ if we consider the tatum-level. Each state in the model generates with some probability an observation vector $\mathbf{O}(t_m)$ at time t_m. This is defined by the observation probabilities. Given the observations, we estimate the most likely chord sequence over time and the downbeat positions in a maximum likelihood sense.

We now describe in detail the characteristics of our HMM: initial state distribution, observation probability distribution and state transition probability distribution.

5.4.3 Initial State Distribution π

The prior probability π_{ik} for each state is the prior probability that a specific chord i occurring on pim_k has been emitted. Since we do not know *a priori* which chord the piece begins with and which *pim* the piece starts with, we initialize π_{ik} at $\frac{1}{I*K}$ for each of the $I * K$ states.

5.4.4 Observation Probabilities

The observation probabilities are computed in the following way. Let $P(\mathbf{O}(t_m)|s_{ik}(t_m))$ denote the probability that observation $\mathbf{O}$ has been emitted at time instant t_m given that the model is in state s_{ik}. Let $P(\mathbf{O}(t_m)|c_i(t_m))$ denote the one that has been emitted by chord c_i and $P(\mathbf{O}(t_m)|pim_k(t_m))$ the one that has been emitted given that the chord is occurring on pim_k. We now assume independence between *chord type* (CM, C#M, ..., Cm, ..., Bm) and the position of the chord in the measure. For instance, we consider that in any given song, even if we favor chord changes on $pim = 1$, we do not favor any *chord type*: a D major chord is as likely to occur at the beginning of a measure as a C major chord[2]. The observation probabilities are computed as:

$$P(\mathbf{O}(t_m)|s_{ik}(t_m)) = P(\mathbf{O}(t_m)|c_i(t_m))P(\mathbf{O}(t_m)|pim_k(t_m)) \tag{5.2}$$

5.4.4.1 Observation *pim* Probability Distribution

Equation (5.2) gives the observation probability for state s_{ik} depending on chord c_i and position in the measure pim_k. Here, the *pim* probability distribution $P(\mathbf{O}(t_m)|pim_k(t_m))$ is considered as uniform ($\frac{1}{K}$ for each *pim* in the measure). It is thus a constant multiplication that has no effect on the observation probability for state s_{ik}, which actually depends only on the chord type. We acknowledge that by doing so, we disregard signal information that could inform the downbeat tracking process. The system would benefit from downbeat information extracted from the signal, for instance by combining a rhythmic pattern approach with the proposed one.

5.4.4.2 Observation Chord Symbol Probability Distribution

The probabilities $P(\mathbf{O}(t_m)|c_i(t_m))$ are obtained by computing at each time instant t_m the cosine distance between the observation vectors (the chroma vectors) $\mathbf{O}(t_m)$ and each of the 24 chord templates $\mathbf{CT}_i, i \in [1, 24]$, which are the theoretical chroma vectors corresponding to the $I = 24$ major and minor triads.

$$For \quad i = 1 \dots 24, \quad P(\mathbf{O}(t_m)|c_i(t_m)) = \frac{\mathbf{O}(t_m).\mathbf{CT}_i}{\|\mathbf{O}(t_m)\|.\|\mathbf{CT}_i\|} \tag{5.3}$$

The chord templates are constructed considering the presence of the higher harmonics of the notes, relying on the model used in the context of key estimation in [Gó6b]. The reader is referred to the previous chapter for more details.

[2]This is not strictly correct: for example some chords are more likely to occur than others on strong beats in the piece according to the musical key. We will not take into account these considerations here, they are left for future work.

The 24 values $P(\mathbf{O}(t_m)|c_i(t_m))$ are normalized across components per template such that their components sum to unity.

$$\sum_i P(\mathbf{O}(t_m)|c_i(t_m)) = 1 \tag{5.4}$$

5.4.5 State Transition Probability Distribution

In music pieces, the transitions between chords result from musical rules. Using a Markov model, we can model these rules in the state transition matrix. In this part, we detail the computation of the state transition matrix. We present in part 5.4.5.1 the main idea on which our model relies on. In this chapter, we are interested in estimating the downbeats of music pieces that may have a complex metrical structure. We first consider the sub-problem of finding the downbeats in the case of a constant 4/4 or 3/4 meter. We detail the corresponding transition matrix in part 5.4.5.2. We then extend the proposed transition matrix to more complex metrical structure cases in part 5.4.5.3. The main notations related to the state transition matrix used in this section are listed in Table 5.2.

Table 5.2: List of the main notations related to the state transition matrix.

Variable	Length	Description
T_c	I	State transition matrix of the chord estimation model presented in Chapter 4.
$T/T_3/T_4$	$I * K$	State transition matrix of the chord/downbeat estimation model (takes into account both the chord transitions and their respective positions in the measure).
$T_{pim}/T_{3pim}/T_{4pim}$	K	Represents the probability to transit between two *pim* and between two different chord types.
T'_{pim}	K	Is used when transiting between two *pim* in the case that the chord type does not change (self-transition case).

5.4.5.1 On the Distribution of Chord Changes According to the Metrical Structure

Let T denote the $I * K$-states transition matrix of the model for simultaneous estimation of chords and downbeats. T takes into account both the chord transitions and their respective positions in the measure. The structure of matrix T is illustrated in Figure 5.5 in the case that $K = 4$ *pim* and $I = 24$ chords. The matrix T is derived from an I-state chord transition matrix denoted by T_c. Here we use the chord transition matrix based on music-theoretical knowledge about key-relationships presented in the previous chapter and used in [PP08b]. Note that the model is not restricted to this choice. Another chord transition matrix could have been used.

That main idea of the present model is that we favor chord changes on the beginning of the measures. In a piece of music, chord changes are in general related to the beats. As

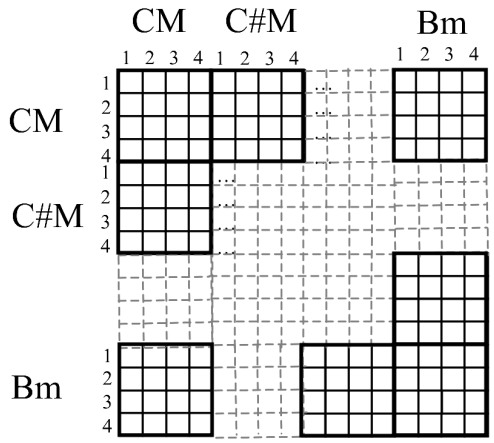

Figure 5.5: Structure of the $I*K$-states transition matrix T used in our model that takes into account both the chord transitions and their respective positions in the measure. Here $K = 4$ *pim* and $I = 24$ chords.

stated by Goto [Got01]:

1. Chords are more likely to change on beat times than on other positions.

2. Chords are more likely to change on half-note times[3] than on other positions of beat times.

3. Chords are more likely to change at the beginnings of measures than at other positions of half-note times.

Analysis of the evaluation test-set has shown that our data support these assumptions. Figure 5.6 shows the distribution of chord changes according to the position in the measure. It can be seen that the three statements reported above are corroborated by the chord annotations. In particular, it can be seen that about 90% of the chord transitions occur on a beat position (for most of them on the strong beats) and 76% of the chord transitions occur on a downbeat.

Because chords are more likely to change at the beginning of a measure than at other *pim*, we give lower self-transition probabilities for chords occurring on the last beat of a measure than on other *pim*. A *self-transition* is defined as a transition between two same chord symbols. For instance CM-CM is a self transition whereas CM-DM is not. The term self-transition here only refers to the spelling of the chord and is independent of its position in the measure.

[3] In pieces in 4/4 meter, the half-note times correspond to temporal positions of strong beats. In other words, a strong beat is either the first or the third quarter note in a measure; a weak beat is either the second or the fourth quarter note in a measure.

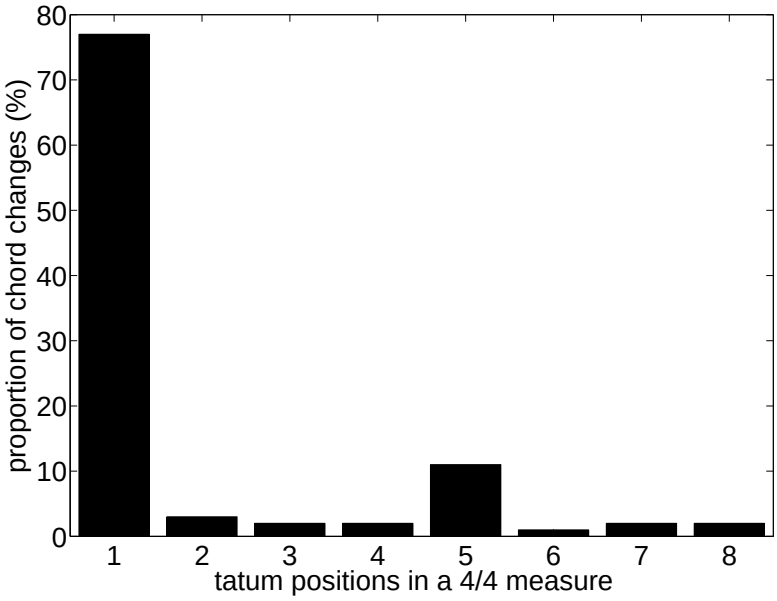

Figure 5.6: Distribution of the chord changes according to the *pim* computed on the pieces in 4/4 of the *Beatles evaluation test-set*.

5.4.5.2 Transition Matrix for a Constant 4/4 Meter

.

Let us consider a simple metrical structure such that the meter is in 4/4 and is constant (this case was first presented in [PP08b]). We also assume that the beat tracking is perfect: there is no insertion or deletion of beats.

We note $T_c(i, i')$ the transition probability between the chords i and i'. This matrix is represented in Figure 5.7 [left].

We also define a *pim* transition matrix T_{pim} which represents the probability to transit from *pim* k to *pim* k'. Since we do not allow our present system to jump over a *pim* (*i.e.* skip over or add one or several beats), only the values $T_{pim}(k, k')$ for $k' = k + 1 \ (mod \ K)$[4] are non-zero. All non-zero values are set to the same value. In case that $K = 4$ (tactus), only the transitions pim_1-pim_2, pim_2-pim_3, pim_3-pim_4, pim_4-pim_1 are thus allowed. The matrix T_{pim} is represented in Figure 5.7 [right, bottom].

We need here to distinguish between two cases:

- the first case concerns transitions between two different chords ($i' \neq i$),
- the second case concerns self-transitions ($i' = i$) and corresponds to the diagonal blocks of T.

[4] where $a = b \ (mod \ m)$ means that a and b have the same remainder for the Euclidian division by m.

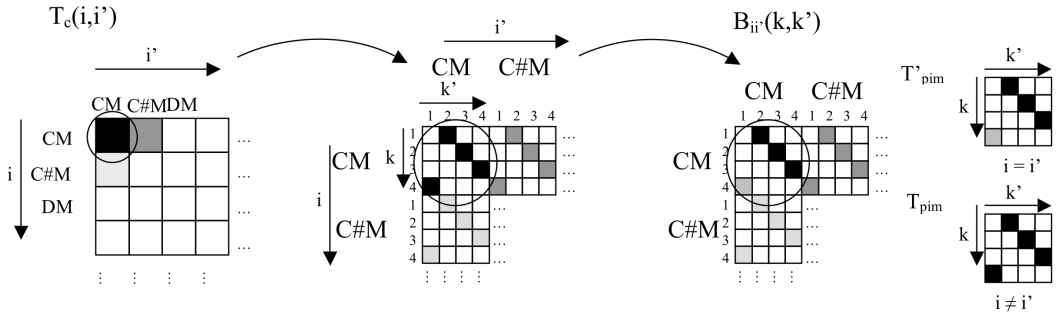

Figure 5.7: Chord transition matrix for a single-state HMM [left], transition matrices for major to major chords in the case of double-states HMM considering a 4/4 constant meter, without taking into account the *pim* of the chord in the measure [middle left] and taking into account the *pim* of the chord [middle right], *pim* transition matrices [right]. In this figure, K = 4. In this figure, the darker the color, the higher the value in the transition matrix.

Since we want to favor chord changes on downbeats, *i.e.* disfavor self-transition between the last *pim* of a measure and the first *pim* of the next measure, we need to define a specific transition matrix for the self-transition case ($i' = i$). This specific matrix is denoted by T'_{pim}. This matrix is represented in Figure 5.7 [right, top].

As one can see T'_{pim} differs from T_{pim} only in the value $T'_{pim}(K,1)$ which is lower than $T_{pim}(K,1)$. The consequence of this lower value is that T'_{pim} disfavors transition between identical chords (self-transition) at measure boundaries. In the opposite T_{pim} does not favor any transition in particular.

From T_c, T_{pim} and T'_{pim}, we construct the global transition matrix T normalized so that the sum of each row is equal to 1 (Figure 5.7 [middle]). Each block $B_{ii'}(k,k')$ of this matrix represents the transition from chord i at *pim* k to chord i' at *pim* k':

$$\begin{cases} B_{ii'}(k,k') & = T_c(i,i') \cdot T_{pim}(k,k') \quad \text{if } i \neq i', \\ & = T_c(i,i') \cdot T'_{pim}(k,k') \quad \text{if } i = i' \end{cases}$$

In the case of a constant 3/4 meter, the transition matrix T can be constructed in the same way except that only transitions pim_1-pim_2, pim_2-pim_3, pim_3-pim_1 are allowed in T_{pim}. The various matrices used for the construction of T are represented in Figure 5.8.

5.4.5.3 Transition Matrix for a Variable Meter

.

We now extend the model presented for constant 4/4 meter to the case of variable meter. In the previous model, the meter was constrained to be constant and it was not allowed to jump over a *pim* (*i.e.* skip over or insert one or several beats). Furthermore, the problem of imperfect beat tracking was not considered.

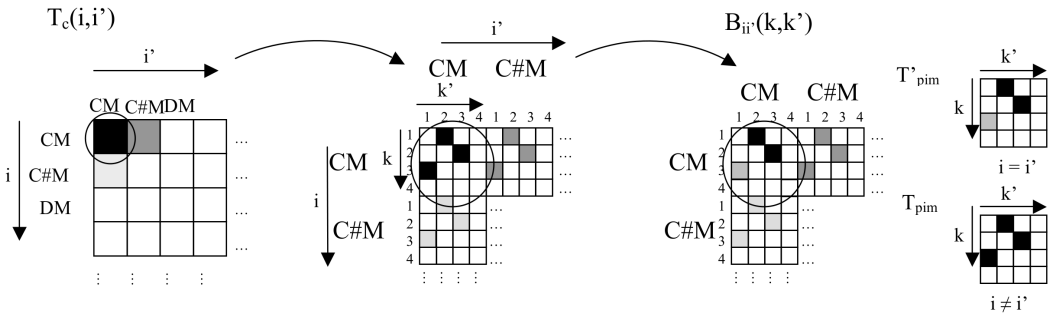

Figure 5.8: Chord transition matrix for a single-state HMM [left], transition matrices for major to major chords in the case of double-states HMM considering a 3/4 constant meter, without taking into account the *pim* of the chord in the measure [middle left] and taking into account the *pim* of the chord [middle right], *pim* transition matrices [right]. In this figure, K = 4. In this figure, the darker the color, the higher the value in the transition matrix.

We now consider the case of variable meter. We will present further in this chapter the results of some experiments that we have carried out on a very large set of Beatles songs presenting various metrical structures. Many of the songs belonging to the test-set do not have a constant meter. Let us recall the several cases of metrical structure considered in our model:

- constant 4/4 or 3/4 meter
- variable meter 4/4 with passages in 3/4 meter
- variable meter 3/4 with passages in 4/4 meter
- insertion of one measure in a different time-signature in a constant 3/4 or 4/4 meter passage.

However, because most of the songs have a predominant meter (3/4 or 4/4), we have chosen to simplify the problem considering two major cases. Two transition matrices, with same form but different values, are proposed. They correspond to the state-transition matrix T described above, in the case of constant 4/4 meter. The first one corresponds to the case of songs in 4/4 meter with ternary passages and will be denoted as T_4. In this case, we favor measures of 4 beats but transitions to measures of 3 beats are allowed. The second transition matrix corresponds to the case of songs in 3/4 meter with passages in 4/4 and will be denoted as T_3. In this case, we favor measures of 3 beats but transitions to measures of 4 beats are allowed. We do not allow the algorithm to skip over or add one or several beats because this would reduce its robustness. Indeed insertion or deletion of beats corresponds to exceptional situations that happen no more than a few times within a song.

In such a situation, the system will probably miss some downbeat positions but it is supposed to catch them up after a few beats. For instance, according to the proposed model, a succession of beats annotated in the ground truth as:

1 2 3 4 1 2 1 2 3 4 1 2 3 4

will theoretically be detected as:

1 2 3 4 1 2 3 1 2 3 1 2 3 4

or as:

1 2 3 1 2 3 1 2 3 4 1 2 3 4

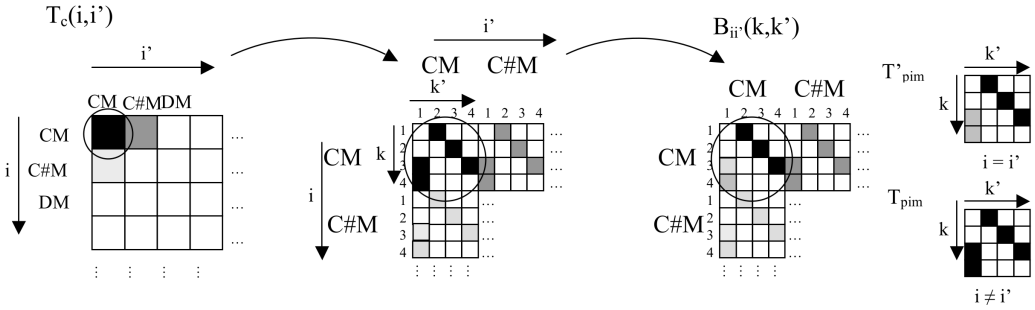

Figure 5.9: In this figure, the darker the color, the higher the value in the transition matrix. The figures indicate: the chord transition matrix for a single-state HMM [left], the transition matrices for major to major chords in the case of double-states HMM, without taking into account the *pim* of the chord in the measure [middle left] and taking into account the *pim* of the chord [middle right] (lower value on transition pim_3-pim_1, pim_4-pim_1) and the *pim* transition matrices [right]. Case of variable meter.

T_3 and T_4 can be seen as block matrices where each block corresponds to a specific chord transition. They are derived from the I-state chord transition matrix T_c presented above. The transition probability between chord i and chord i' will be denoted as $T_c(i, i')$. This matrix is represented in the left part of Figure 5.9.

T_3 and T_4 are related to both the metric and harmonic structures of a piece of music. The construction of T_3 and T_4 follows three steps. The first two concern the problem of the downbeats. The third step takes into account the chord type dimension.

Firstly two *pim* transition matrices T_{3pim} and T_{4pim} are defined, which represent the probability to transit from pim_k to $pim_{k'}$ in a song. According to our assumptions, only values $T_{3pim}(k, k')/T_{4pim}(k, k')$ such that $k' = (k + 1)\ (mod\ 4)$[5] are non-zero, as well as $T_{3pim}(3, 4)/T_{4pim}(3, 4)$ so that transitions between measures in 4/4 and measures in 3/4 are allowed:

[5]where $a = b\ (mod\ m)$ means that a and b have the same remainder for the Euclidian division by m.

$$\left\{\begin{array}{rl} T_{3pim}(1,2) = & 1 \\ T_{3pim}(2,3) = & 1 \\ T_{3pim}(3,4) = & \alpha_3 \\ T_{3pim}(4,1) = & 1 \\ T_{3pim}(3,1) = & \beta_3 \end{array}\right. \left\{\begin{array}{rl} T_{4pim}(1,2) = & 1 \\ T_{4pim}(2,3) = & 1 \\ T_{4pim}(3,4) = & \alpha_4 \\ T_{4pim}(4,1) = & 1 \\ T_{4pim}(3,1) = & \beta_4 \end{array}\right. \tag{5.5}$$

with $\alpha_3 < \alpha_4$ and $\beta_3 > \beta_4$ so that measures in 3/4 are favored in the case of T_{3pim} and measures in 4/4 are favored in the case of T_{4pim}. In our experiments, we used $\alpha_3 = 0.6$, $\alpha_4 = 0.9$, $\beta_3 = 1.05$ and $\beta_4 = 0.85$. These values were manually selected in small scale simulations, starting from the value 1 and varying in a range of ± 0.05.

Secondly, we want to favor chord changes on downbeats, *i.e.* disfavor transition between identical chords at measure boundaries (between the last *pim* of a measure and the first *pim* of the next measure). For this self-transition case ($i' = i$), corresponding to the diagonal blocks of T_3 and T_4, we define a specific transition matrix, denoted by T'_{pim}. T'_{pim} is the same in the case of T_3 building and in the case of T_4 building. To favor chord changes on downbeats, we attribute a self-transition probability from beat 3 to beat 1 (3/4 time-signature) and from beat 4 to beat 1 (4/4 time-signature) lower than on other *pim* transitions:

$$\left\{\begin{array}{rl} T'_{pim}(1,2) = & \alpha \\ T'_{pim}(2,3) = & \beta \\ T'_{pim}(3,4) = & \gamma \\ T'_{pim}(4,1) = & \delta \\ T'_{pim}(3,1) = & \delta \end{array}\right. \quad \text{s.t.} \quad \delta < \alpha, \beta, \gamma \tag{5.6}$$

The values α, β, γ and δ were again selected manually in small-scale simulations starting from the value 1 and varying in a range of ± 0.05, testing values between 0.5 and 1.5.

It should be noted that, even if the model parameters were selected in part by hand and have an impact on the results, the exact values of these parameters is not critical. The same transition matrices with the same parameters have been used with success on a set of classical music pieces [PP09], which suggest that the values are not critical to the test-set. It should be possible to learn the parameters from the annotated files. However, until now, our attempts to derive transition probabilities from the training set have given less accurate results than those obtained using values selected in part by hand.

Finally, we construct the global transition matrix T_3 from T_c, T_{3pim} and T'_{pim}, and normalize it so that the sum of each row is equal to 1 (Figure 5.9 [middle]). Each block $B_{ii'}(k,k')$ of this matrix represents the transition from chord i at pim_k to chord i' at $pim_{k'}$:

$$B_{ii'}(k,k') = \left\{\begin{array}{cl} T_c(i,i') \cdot T_{3pim}(k,k') & \text{if} \quad i \neq i' \\ T_c(i,i') \cdot T_{3pim}(k,k') \cdot T'_{pim}(k,k') & \text{if} \quad i = i' \end{array}\right. \tag{5.7}$$

The transition matrix T_4 is constructed in a similar way, using T_c, T_{4pim} and T'_{pim}.

5.4.6 Simultaneous Estimation of Chords and Downbeats

In order to find the optimal succession of states s_{ik} over time, the Viterbi decoding algorithm [GM99a] is used successively with the two chord transition matrices T_3 and T_4. The algorithm provides the most likely path $\mathbf{Q}$ through the HMM states given the sequence of observations. The transitions matrix T which gives the greatest likelihood given the observation sequence $\mathbf{O}$ according to Equation (5.8) is selected. We obtain simultaneously the best sequence of chords over time and the downbeat positions.

$$T = \text{argmax}(P(\mathbf{O}, \mathbf{Q}|T_3), P(\mathbf{O}, \mathbf{Q}|T_4)) \tag{5.8}$$

5.5 Evaluation Method

The proposed model is tested on a subset of 165 songs of the *Beatles test-set*. The test-set includes pieces in 3/4, 4/4 and variable meter. We refer the reader to Chapter 2, sections 2.3.2 and 2.5 for more details about this test-set.

To assess the performance of the system, we use the evaluation measures described in Chapter 2, sections 2.4. Beat and downbeat tracking are evaluated using the standard Precision, Recall and F-measure. We consider two aspects of chord estimation: the label accuracy (how the estimated chord is consistent with the ground truth) and the segmentation accuracy (how the detected chord changes are consistent with the actual locations).

5.6 Analysis of the Results

We provide in this section a detailed analysis of the results. We illustrate it using some examples that have been chosen for their relevance. Since the interrelationship between musical attributes is the main purpose of this work, special attention is devoted to this aspect. This section starts with a global presentation of the results. We then analyze in detail the downbeat estimation results. We continue with a comparison of the chord estimation results with other state-of-the-art chord detection systems through the Music Information Retrieval Evaluation eXchange (MIREX) 2008 and 2009 results. This comparison is followed by a discussion about the influence of each musical attribute on the estimation of the other. We finish with some case study examples.

5.6.1 Chords and Downbeats Interaction

The results are presented in Tables 5.3, and 6.5 and illustrated in Figure 5.10. A downbeat tracking accuracy result up to 79% (EB-TAT) suggests that relying on the chord structure of a piece is an appropriated approach for downbeat estimation. Conversely,

Table 5.3: Chord label accuracy results (in %) considering several cases: not integrating/integrating metrical structure information in the transition matrix of the model (NM/WM), tactus or tatum synchronous analysis (TAC/TAT), using theoretical beat positions (TB) or automatically estimated beat positions (EB). Rel. Imp. and Stat. Sig.: relative improvement and statistical significance between the cases NM and WM.

no meter (NM)			
theoretical beats (TB)		estimated beats (EB)	
TAC	TAT	TAC	TAT
69.6 ± 13.9	71.2 ± 13.1	68.5 ± 14.0	71.2 ± 13.1

with meter (WM)			
theoretical beats (TB)		estimated beats (EB)	
TAC	TAT	TAC	TAT
71.5 ± 13.3	72.9 ± 13.3	70.4 ± 14.2	72.8 ± 13.3

	TAC	TAT	TAC	TAT
Rel. Imp. WM/NM (%)	$\frac{71.5-69.6}{69.6} = 2.7$	2.4	2.8	2.2
Stat. Sig.	yes	yes	yes	yes

Table 5.4: Downbeat position estimation results considering several cases: theoretical or estimated beats (TB/EB), tactus/tatum-synchronous analysis (TAC/TAT). Precision (Prec), Recall (Rec), F-measure (F-m).

	theoretical beats (TB)		estimated beats (EB)	
	TAC	TAT	TAC	TAT
Prec	0.89 ± 0.20	0.84 ± 0.26	0.76 ± 0.30	0.80 ± 0.26
Rec	0.90 ± 0.20	0.86 ± 0.26	0.76 ± 0.31	0.79 ± 0.28
F-m	0.89 ± 0.20	0.85 ± 0.26	0.76 ± 0.31	0.79 ± 0.27

Table 5.5: Beat position estimation results.

Precision	Recall	F-measure
0.91 ± 0.22	0.88 ± 0.24	0.89 ± 0.23

taking into account the metrical structure allows us to improve the chord recognition task by 2.8% relative improvement in the case of tactus-frame analysis and 2.2% relative improvement in the case of tatum-frame analysis.

We performed a paired sample t-test to determine if there is a significant difference between the chord estimation results obtained without considering interaction with the metrical structure (NM) and with consideration of interaction with the metrical structure (WM). For the various situations (TB, EB, TAC, TAT), the null hypothesis could be rejected at the 5% significance level. We can conclude that there is a statistical difference on the chord estimation results when considering the metrical structure in the model.

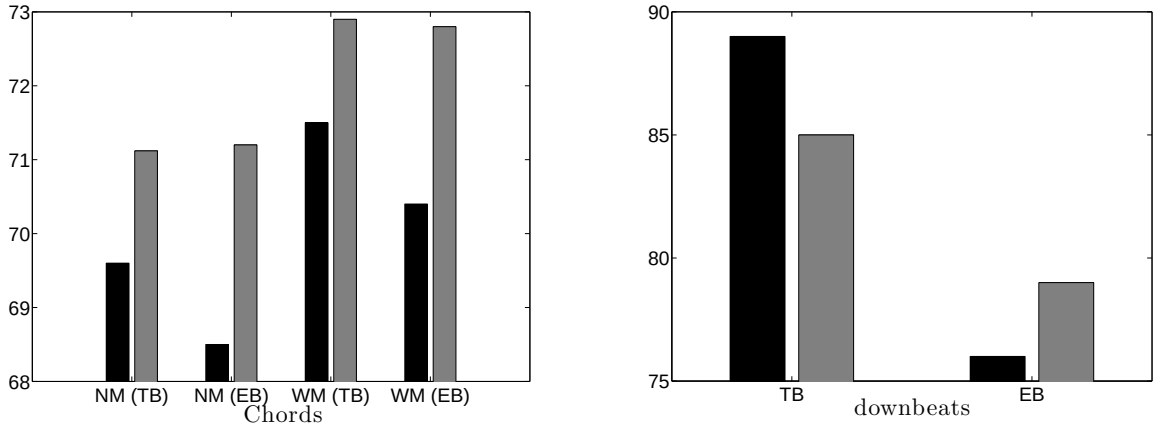

Figure 5.10: Histogram of chord [left] and downbeat [right] estimation results (in %) considering several cases: not integrating/integrating metrical structure information in the model (NM/WM); using theoretical beat positions (TB) or automatically estimated beat positions (EB). The results from the tactus-synchronous analysis are represented in black, the results from the tatum-synchronous analysis are represented in grey.

5.6.2 Downbeat Position Estimation

In this section, we evaluate the performance of downbeat estimation comparing the output of our algorithm to the ground truth downbeat times annotated by hand. Following the approach proposed in [Dav07], we measure the performance of downbeat estimation considering two cases. On the one hand, we evaluate the upper limit of the model by estimating the downbeat positions using manual annotation of beat positions (referred to as theoretical beat positions (TB) in Table 6.5). On the other hand, we measure the fully automatic performances of the system by using a beat tracker [Pee] as a front end of the system. The beat positions estimated with the beat tracker are referred to as estimated beat positions (EB) in Table 6.5. With these two measures, we can distinguish between errors due to poor beat position estimation and errors due to the model.

5.6.2.1 Semi-automatic Downbeat Position Estimation

The results presented in Table 6.5 show that the system leads to a good estimation of downbeat positions. It achieves 89% of correct estimation in the case of tactus-synchronous analysis and 85% in the case of tatum-synchronous analysis. The encouraging results obtained in the case of tatum-synchronous analysis highlight the robustness of the presented approach.

It can be remarked that the standard deviation is high. This can be explained by the fact that the downbeat estimation score is null for some pieces, in particular when there

are many half-measure chord changes in the song. In this case, the downbeat positions may be located by the algorithm on the third instead of the first beats of the measures.

5.6.2.2 Using Estimated Beat Positions Versus Theoretical Beat Positions

The downbeats estimation relies on the knowledge of the beat positions. For real applications of the system, we need to use automatically estimated beats. Errors in the beat tracking will be carried forward into the downbeat tracking stage. Beat tracking results evaluated on the test-set are presented in Table 5.5 and show that the beat tracking is not perfect. We thus expect a lower downbeat tracking performance using the estimated beats compared to downbeat tracking performance using the ground-truth beats. However, the decrease in the results from semi-automatic to fully-automatic analysis is lower in the case of tatum-frame analysis than in the case of tactus-frame analysis because some common beat tracking errors do not affect downbeat estimation at the tatum-level.

Some common errors in beat tracking algorithm are octave errors (*e.g.*, halving or doubling the beat positions). In case of halved beat positions, a maximum downbeat tracking score recall of 0.5 using tactus-frame features can be expected, but a recall of 1 could be theoretically reached using tatum-frame features.

Off-beat errors (tapping at the annotated metrical on the off-beat positions) in addition to a halved tempo estimation is another common beat tracking error. If such a beat estimation error is constant throughout the analyzed piece, we expect to have a null score for tactus-synchronous analysis but a score similar to the one obtained using the theoretical beat position for tatum-synchronous analysis. This was corroborated in our experiments.

An interesting case of beat estimation errors concerns the insertion or deletion of beats due to a tempo deviation (*e.g.*, slowdown in the tempo). The presented system is supposed to tackle this situation as it does when there is beat insertion or deletion within the music (see below).

5.6.2.3 Comparison With the State-of-the-art

We compare the performance of our algorithm (WM-EB-TAT) against those obtained using M.E.P. Davies's model [Dav07], which we refer to as MEPD. Many thanks to Matthew E. P. Davies for making his source code available. In the MEPD approach, the downbeats are estimated based on spectral difference between band-limited beat synchronous analysis frames. The analysis is restricted to the cases where the time signature does not change. The algorithm requires a sequence of beat times and the time-signature of the input signal to be known a priori. For comparison with our system, we have used our beat tracker as input to the MEPD downbeat estimation system. We computed i) the results across the whole test-set, ii) the results across the songs for which the beat tracking was perfect, iii)

the results across the songs for which the beat tracking was imperfect.

Table 5.6: Downbeat estimation results for proposed approach (PA), MEPD approach (MEPD). Results across the whole test-set (Whole Data), results across songs with perfect beat tracking (Perfect BT), results across songs with imperfect beat tracking (Imperfect BT). Precision (Prec), Recall (Rec), F-measure (F-m).

	Whole Data	
	MEPD	PA
Precision	0.74 ± 0.36	0.81 ± 0.26
Recall	0.72 ± 0.37	0.79 ± 0.28
F-measure	0.72 ± 0.36	0.79 ± 0.27

	Perfect BT		Imperfect BT	
	MEPD	PA	MEPD	PA
Prec	0.90 ± 0.24	0.86 ± 0.26	0.36 ± 0.30	0.71 ± 0.24
Rec	0.90 ± 0.24	0.87 ± 0.26	0.30 ± 0.25	0.62 ± 0.25
F-m	0.90 ± 0.24	0.86 ± 0.26	0.32 ± 0.25	0.64 ± 0.23

Results reported in Table 5.6 show that our system is globally more successful than the MEPD approach and thus compares favorably to the state-of-the-art. MEPD obtains better results across songs with perfect beat tracking. Most of those songs have a constant time-signature. For those files, The MEPD accuracy for each of those files will either be 0 or 100%, whereas our system may insert some additional downbeats. This highlights a shortcoming of our system: we need to make a compromise between favoring constant meter and allowing meter changes (see below). However, our system performs clearly better across the songs on which the beat tracking was imperfect. Any added or omitted beat in the beat tracking will irrecoverably degrade the MEPD downbeat tracking process whereas our system can handle those situations. Our system thus shows improvements over the state-of the-art.

5.6.2.4 Handling Variable Meter

Previous works on downbeat tracking have mostly focused on pieces with constant meter. The present work proposes an approach that considers some cases of variable meter. The results we obtain are encouraging. We obtain a score of 56% on tactus and tatum analysis on the 9 variable meter pieces of the test-set. For each song, we can determine a predominant meter. The transition matrix corresponding to the predominant meter of the piece has been correctly chosen for all songs but one. Ideally, the system should remain in the new meter when a change in meter occurs. However, the model is built in order to favor constant meter within a music piece. For this reason, if the chord changes are not strongly enough marked in the chromagram (high spectral difference between frames), the system will not adapt to the meter change until there is a sufficiently clear chord change. In case of a meter change from a predominant meter 3/4 to 4/4, the proposed algorithm inserts measures in 4/4 so that most of the downbeat positions are correctly detected when the predominant meter returns.

Let us illustrate this on an example, Figure 5.11. The song *I Me Mine* has a 3/4 predominant meter with changes to 4/4 meter. Due to percussive sounds, the chromagram is blurred and chord changes are not clear. Note that the beat positions are not perfectly estimated by the beat tracker (see the dashed rectangle). It can be seen in Figure 5.11 that, during the 4/4 meter passage (from $33s$ to $55s$), the system mostly remains in 3/4. However, measures in 4/4 are inserted (see the black circles) so that the downbeats are correctly estimated when the 3/4 meter returns. Our model shows some adaptation to meter changes even if it is not perfect. On the provided example, the 11 bars of the 4/4 section cannot be divided into a whole number 3/4 bars. If the system had constantly remained in 3/4, the rest of the downbeats until the end of the song wouldn't have been correctly detected. Note that, even for many human listeners, it is very difficult to understand meter changes on this complex example. Experiments carried out by the author on 6 trained musicians clapping their hands along with the music have shown that listeners needed between 2 and 3 measures before synchronizing with the correct downbeat positions of the 4/4 meter passage.

The last line in Figure 5.11 represents the downbeat tracking obtained by increasing the value of α in Eq.(5.5), so that constant 3/4 meter is less favored by the model. With this value, the algorithm shows more flexibility to the meter change. We plan to find methods to reduce the trade-off between favoring constant meter and allowing meter changes.

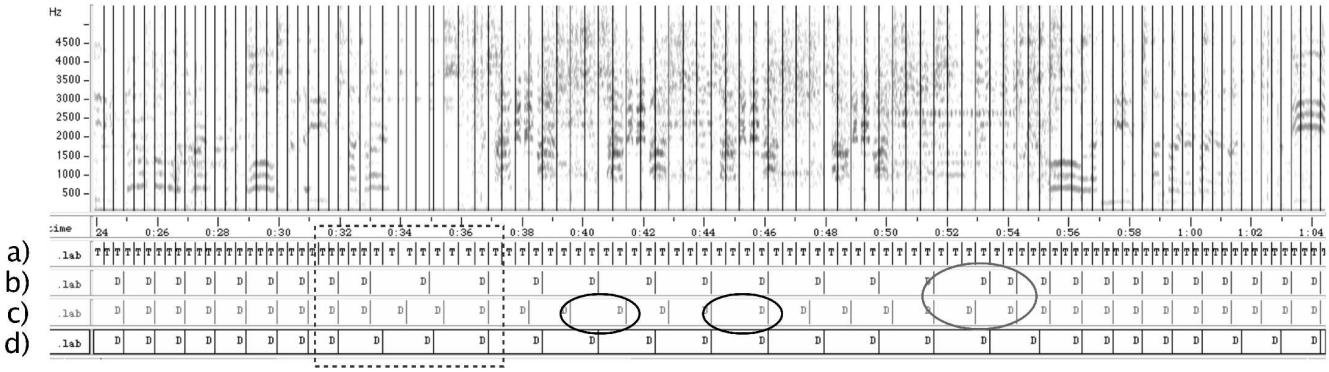

Figure 5.11: Estimated downbeat positions of an excerpt of the song *I Me Mine*. Annotated beat positions [top, a)], annotated downbeat positions [middle top, b)], estimated downbeat positions [middle bottom, c)] with $\alpha_3 = 0.6$, estimated downbeat positions [middle bottom, d)] with $\alpha_3 = 0.85$. Measures in 4/4 inserted by the model are indicated by the two black circles. Extra beats added by the Beatles at the end of the passage in 4/4 meter are indicated by the grey circle. The dashed rectangle shows a region with errors in the beat tracking. The image has been obtained using the Open Source tool *Wavesurfer*.

5.6.2.5 Handling Insertion or Deletion of Beats

It is possible that there is an insertion or an omission of beats within a constant-meter part of a song, either due to the music itself or due to a beat tracking error. This is illustrated in Figure 5.11: two extra beats have been added by the Beatles at the end of the passage in 4/4 meter (see the grey circle). The estimated succession of beats is 1 2 3 1 2 3 1 2 3 instead of 1 2 3 4 1 2 1 2 3. This corroborates our expectations stated in part 5.4.5: the system synchronizes to the correct downbeat positions after a few beats following the added or deleted beat.

5.6.3 Chord Estimation

In this section, we analyze the performances of chord label estimation comparing the output of our algorithm to ground truth chord labels annotated by hand.

5.6.3.1 Comparison of the Results with MIREX 2008 "Audio Chord Detection"

We participated to the first chord detection task in Music Information Retrieval Evaluation eXchange[6]. In the submitted system, the chords were estimated without considering interaction with downbeats. To set the algorithm presented in this chapter among other state-of-the-art chord detection algorithms, we first report and analyze the MIREX 2008 chord detection results.

The MIREX 2008 Audio Chord Detection task was divided into two subtasks. In the first subtask the systems were pre-trained and tested against 176 Beatles songs. In the second subtask systems were trained on 2/3 of the Beatles test-set and tested on 1/3. Our system does not need any training, we thus participated to the first subtask. An overlap score was calculated as the ratio between the overlap of the ground truth and detected chords and ground truth duration. Four songs were excluded from the original Beatles test-set because of problems aligning the ground truth chords to the audio data.

A total of 8 algorithms were submitted to the first subtask, and our algorithm obtained the fourth place. The various results are reported in Figure 4.1, chapter 4. Note that silent or no-chord segments were not estimated with our algorithm. The differences in the results between the participants are rather small, probably because the approaches are similar (using HMM). The four highest results were the following: Bello and Pickens [BP05] obtained 66% of correct detected chords, Mehnert [MGAZ08] 65% correct, Ryynänen and Klapuri [RK08a] 64% correct, Papadopoulos and Peeters [PP08a] 63% correct. Our system compares favorably to the trained-systems. Indeed, 7 algorithms were submitted to the second subtask. The approach proposed by Uchiyama, Miyamoto, and Sagayama [UMOS08] gave results that were significantly better than the other

[6] http://www.music-ir.org/mirex/2008/

submitted algorithms (72% correct). Ellis obtained [Ell08] 66% correct results. All the remaining algorithms gave results above 62%.

Using MIREX's exact methodology (chord evaluation measure and test-set), we have re-computed the score obtained with our MIREX 2008 algorithm and computed the score obtained with the newly proposed algorithm (EB-WM-TAT) on the 162 Beatles songs of our test-set. We obtained a statistically significant relative improvement of 2.4%.

Note that despite using MIREX methodology, we did not recover the results reported by MIREX. We obtained 68.8% for our MIREX algorithm and 70.5% for the newly proposed algorithm. A deeper analysis of MIREX's results has shown that it is very likely that there are some errors in the evaluation. For instance, all participants obtained a score close to zero for songs number 23, 35, 76, 97 and 104.

5.6.3.2 Comparison of the Results With MIREX 2009 "Audio Chord Detection"

We submitted the present system to the second chord detection contest in Music Information Retrieval Evaluation eXchange 2009[7]. The MIREX 2009 audio chord detection task description is similar to the one proposed in 2008 except that the score computation is slightly different. A first score is calculated as the ratio between the overlap of the ground truth and detected chords and ground truth duration, then a weighted average is computed across the songs by weighting each score by the song duration. In 2009, the test-set included also 37 popular music songs. A total number of 13 algorithms were submitted. The chord accuracy results are represented in Figure 4.2 in Chapter 4.2. They vary between 53.8% and 71.2%. Our algorithm ranked 8^{th} with a result of 63.2%.

It is interesting to note that that our algorithm (PP) and two other algorithms submitted to the MIREX contest, including the second best MIREX algorithm (OGF2), were compared against each other in the framework of the QUAERO project[8]. The algorithms were evaluated on a database of 20 popular music songs that did not include any Beatles songs. The results are represented in Figure 5.12. The OGF2 evaluated for the QUAERO project had been pre-trained on the *Beatles test-set*, as for MIREX 2009. Our algorithm obtained a score of 73.18% (PP) and it outperformed (OGF1) which obtained a score of 72.55%, although (OGF1) had outperformed our algorithm in MIREX 2009.

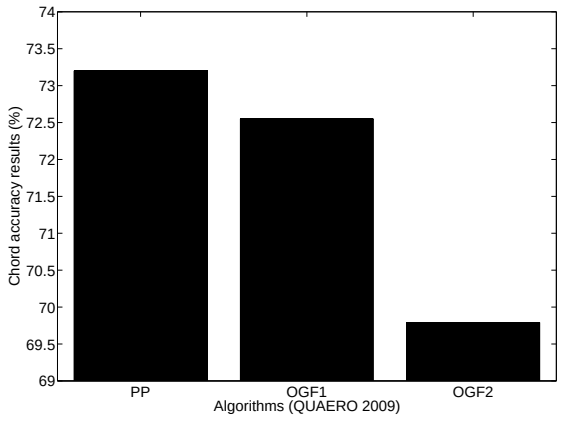

Figure 5.12: QUAERO 2009 Audio chord detection Results: PP [Papadopoulos and Peeters] - OGF1 [Oudre *et al.* 1 (majmin)] - OGF2 [Oudre *et al.* 2 (majmin)].

Table 5.7: Chord segmentation accuracy results (in %) considering several cases: not integrating/integrating metrical structure information in the model (NM/WM), tactus or tatum analysis (TAC/TAT), using theoretical beat positions (TB) or automatically estimated beat positions (EB). Rel. Imp. (%) indicates the relative improvement between the two approaches. Precision (Prec), Recall (Rec), F-measure (F-m).

	no meter (NM)			
	theoretical beats (TB)		estimated beats (EB)	
	TAC	TAT	TAC	TAT
Prec	61.6 ± 16.8	43.7 ± 15.0	55.6 ± 21.4	43.6 ± 15.1
Rec	59.1 ± 17.1	56.5 ± 16.7	52.7 ± 21.3	56.4 ± 16.6
F-m	59.0 ± 15.2	48.4 ± 15.0	52.8 ± 19.8	48.2 ± 14.8

	with meter (WM)			
	theoretical beats (TB)		estimated beats (EB)	
	TAC	TAT	TAC	TAT
Prec	68.3 ± 17.7	57.4 ± 18.0	61.3 ± 23.2	56.8 ± 18.3
Rec	72.5 ± 17.1	73.5 ± 18.4	64.4 ± 23.8	71.1 ± 18.7
F-m	69.1 ± 15.8	63.3 ± 17.2	61.6 ± 22.1	62.0 ± 17.2
Rel. Imp.	17.12	30.8	16.7	28.6

5.6.3.3 Chord Segmentation

Table 5.7 presents the chord segmentation accuracy results. It can be seen that jointly estimating the downbeats with the chords improves significantly the chord segmentation. Chord estimation results presented in Table 5.3 may seem contradictory since, in the TB case, tatum-based features result in better chord estimation whereas tactus-based features result in better downbeat tracking. It is worth noting that chord estimation is better on tatum-based results even without joint estimation of chords/downbeats. This may be explained by the fact that tatum-based analysis takes chord changes on off-beats into account, whereas tactus-frame analysis only allows chord changes on beats. However, the improvement of the chord segmentation accuracy is a consequence of the improvement of the downbeat estimation accuracy. For instance, downbeat estimation based on the beat tracking is better on tatum-frame analysis than on tactus-frame analysis and consequently, chord segmentation is slightly better on tatum-frame analysis than on tactus-frame analysis.

5.6.3.4 Analysis of Chord Detection Errors

In this part, we focus only on chord estimation results and analyze chord detection errors. The results indicated in Table 5.3 show that we obtain up to 72.8% of correctly identified chords on our test-set. As can be seen, the standard deviation of the results is relatively high (around 13%). Analysis of the chord estimation errors leads to similar conclusions than the ones obtained in Chapter 4. The most common errors correspond to neighboring triad confusion and are confusions due to ambiguous chord lexicon mapping, passing tones or missing notes.

Table 5.8: Proportion (in%) of chord errors corresponding to harmonically related chords.

Relative	Dominant	Sub-dominant	Parallel
10%	13%	34%	13%

Table 5.8 shows that a large portion of chord errors (about 57%) corresponds to harmonically close triad confusion: relative chords (Am being confused with CM); dominant chords (CM being confused with GM) or subdominant chords (CM being confused with FM). Parallel major/minor chords (EM being confused with Em) account for 13%. The distribution of the type of errors is similar for all the configurations of the system (TAC, TAT, TB, EB). Note that there is a notable predominance of sub-dominant errors in the results. This may be due to the high value given to transitions between subdominant chords in the cognitive-based transition matrix. We have found that diminishing this

[7] http://www.music-ir.org/mirex/2009/

[8] Quaero is a collaborative research and development program focusing on the areas of automatic extraction of information, analysis, classification and usage of digital multimedia content for professionals and consumers. IRCAM is in charge of the coordination of audio/music indexing research and of development of music-audio indexing technology. A specificity of the project is the creation of a large-music-audio corpus in order to train and validate all the algorithms developed during the project.

value decreases the sub-dominant errors rate. However, this also decreases the global results. This shows some limitations of our approach that is not based on training but on theoretical and cognitive-based music knowledge. If the system does not recognize exactly a chord but makes such confusions, the result can still be useful for higher-level structural analysis such as key estimation, harmony progression or segmentation. The results obtained by the system when taking into account these harmonically close chords are quite high (80%).

The harmonically close chord errors do not have all the same qualitative weight. Parallel errors, for instance, may badly affect key recognition. However, the most common harmonically close errors are dominant and sub-dominant chords (having a perfect fifth relationship between the estimated and ground-truth chord), which should not affect key estimation. Neighboring triad confusion may not be critical to downbeat estimation. A relevant example of this assessment here concerns the detection of metrical structure. We obtain a score of 57% of correctly detected chords on the song *Don't Bother Me*, which is rather low compared to the other songs. However, most of the errors correspond to neighboring chords and the harmonic structure has been well-preserved (chord changes occur according to the measures), as illustrated in Figure 5.13. For that reason, the downbeat positions of the song have been correctly detected.

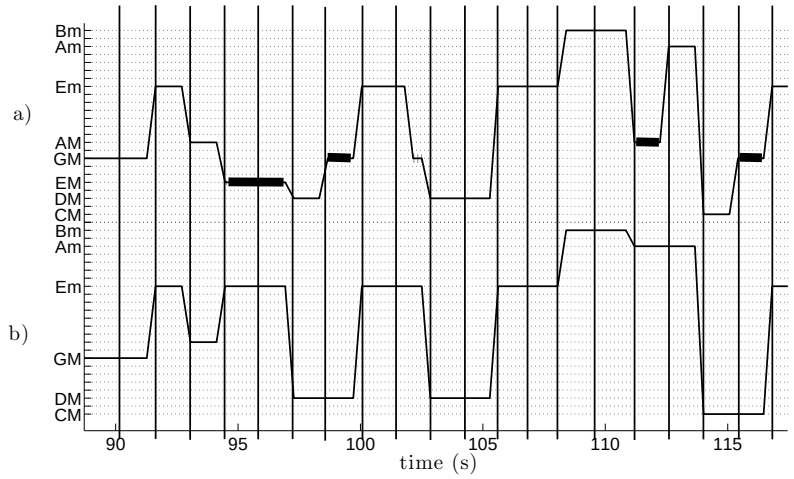

Figure 5.13: Estimated chord progression of an excerpt of the song *Don't Bother Me* [a)] and ground truth [b)]. The downbeat positions are represented by the vertical lines.

5.6.3.5 Tactus-synchronous Versus Tactum-synchronous Analysis

Table 5.3 indicates that the tatum-frame analysis performs slightly better in general than the tactus-frame one. This may be due to the fact that tatum-based analysis takes chord changes on off-beats into account, whereas tactus-frame analysis only allows chord changes on beats.

5.6.4 Case Study Examples

In this part, we present some examples that illustrate some important advantages when estimating simultaneously the chords and the downbeat positions.

5.6.4.1 Boundary Errors

Taking into account the position of the downbeats when estimating the chord progression allows us to improve the accuracy of the estimation. Indeed, when this information is not considered, the chord change may be detected a beat before or after its theoretical position, because of the smoothness of chord transition. This is illustrated in Figure 5.14. When the chords are estimated independently of downbeat positions, errors often occur around *pim*. When the downbeat positions are taken into account, chord changes on the correct position are favored (see line [b]).

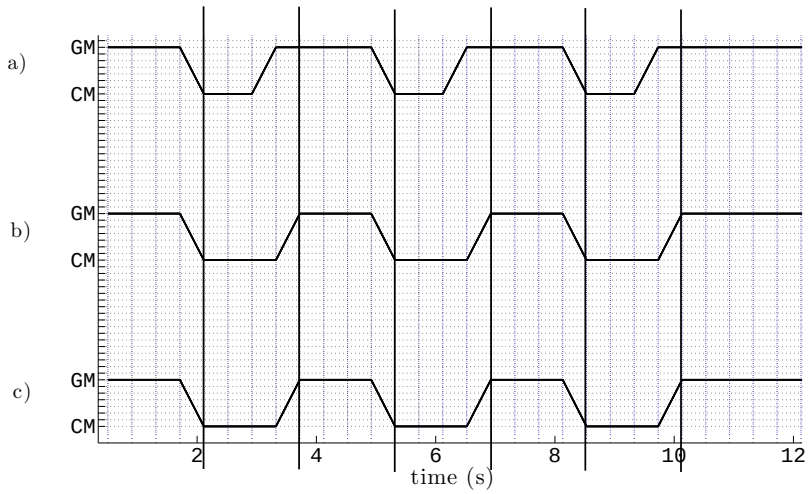

Figure 5.14: Chord progression of the first few seconds of the song *Love Me Do* without taking into account the downbeat positions [a] and taking into account the downbeat positions [b]. Ground truth [c]. The downbeat positions are represented by the vertical lines.

5.6.4.2 Chord Changes

The example in Figure 5.15 clearly shows how the chord progression estimation task can benefit from modeling chord dependencies to the metrical structure. This piece is in CM key and it changes between CM and GM chords about every two measures [ground-truth line c)]. Without taking into account global dependencies [line a)], chord transitions are badly detected and the estimated chord progression remains almost all the time on the GM chord instead of transiting between GM and CM. The knowledge of downbeat positions

[line b)] allows us to better detect transitions.

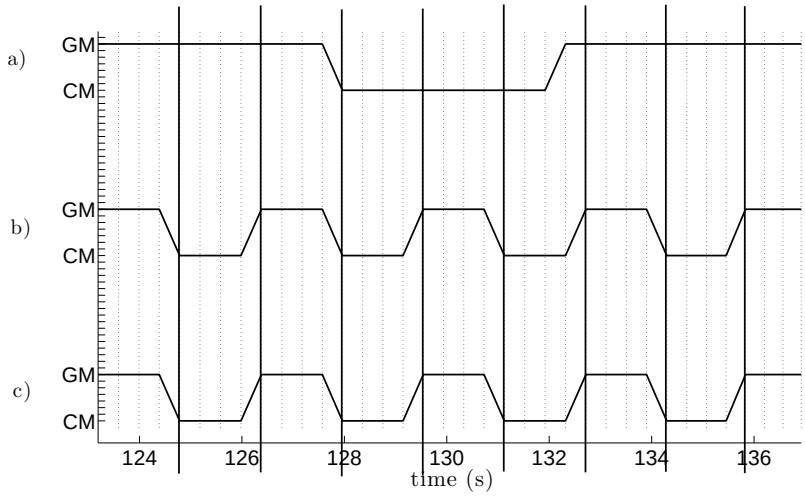

Figure 5.15: Chord progression of the last few seconds of the song *Love Me Do* without taking into account the downbeat positions [a] and taking into account the downbeat positions [b]. Ground truth [c]. The downbeat positions are represented by the vertical lines.

5.7 Conclusion

In this chapter, we have presented a model that allows the simultaneous estimation of the chord progression and the downbeat positions of an audio file. The key idea behind our approach is that the harmonic structure is closely related to the metrical structure of a piece of music. Relying on this idea, we have built a specific topology of HMM where each state is a combination of an occurrence of a chord and a position of the chord in the measure. Each state is thus related on the one hand to the harmonic structure and on the other hand to the metrical structure of the piece. Harmonic structure information and metrical structure information are encoded in the state transition matrix. The chord progression and the downbeats are estimated jointly based on the assumption that chords are more likely to change at the beginning of a measure than on other positions. We consider that an important contribution of our work is that we treat the case of pieces with varying time-signatures.

The system has been evaluated and compared to the state-of-the-art on a large set of hand-labeled files. We have demonstrated that considering the interaction between the two musical attributes allows their simultaneous estimation and that the robustness and the chord estimation accuracy is higher when estimated jointly with downbeats.

We have provided a detailed analysis of the results illustrated by case studies that suggest that some points need further improvement such as a pre-processing step that removes transients and noise and the use of longer dependencies between chords (using, for instance, probabilistic N-grams).

We have considered the problem of using imperfect beat positions obtained from beat tracking. Results show that using a tatum-synchronous analysis instead of a tactus-synchronous one might temper the effects of imperfect beat tracking on downbeat tracking. The model allows us to take into account pieces with complex metrical structure. The downbeat tracking results for pieces in variable meter are encouraging even if they need further improvement. For the moment, the system is built so that it remains in general in a single predominant meter along the analyzed track. It would be highly desirable that the system shows more flexibility to the meter changes. Future work will concentrate on this point.

For the moment, the model has mainly been tested on popular music. We plan to incorporate a larger test database to explore the performances of the system on various types of music genres. We also plan to quantify the downbeat performances of the system with more elaborated measures.

It should be noticed that, for some reasons explained in the previous chapter, we have restricted our chord lexicon to the 24 major and minor triads. We think that the proposed model for joint estimation of chords and downbeats could be directly extended to a larger chord lexicon. However, we do not have performed experiments to corroborate this claim and we left this point for future works.

An analysis of the results shows that the harmonic structure of a piece is an important clue for determining the downbeat positions. However, it has been noticed that in some cases (such as when chords change every two beats), the relationship between chord changes and downbeats is ambiguous. This model would benefit from a more complete functional chord analysis. Combining the present system, which is based on harmony, with a rhythmic pattern approach would probably also allow improvement of the downbeat tracking process.

Chapter 6

Interaction Between Chords, Downbeats and Keys

In this chapter, we present our model for simultaneous downbeat, chord and key estimation from an audio signal. Because the musical key is deeply related to the chord progression and the metrical structure, their automatic estimation should be improved by exploiting their interrelationship.
We first focus on the problem of global key estimation. Relying on previous works on key estimation, we extend the model presented in the previous chapter by integrating global key information and considering interaction between chords, downbeats and the musical key.
We then draw our attention to the problem of local key finding. We propose to address the problem of local key finding by investigating the possible combination and extension of different previous proposed global key estimation approaches. The specificity of our approach is that we introduce key dependency on the harmonic and the metrical structures.
A contribution of our work is that we address the problem of finding a good analysis window length for local key estimation by introducing information related to the metrical structure in our model. Key estimation is not performed on empirical-chosen segment length but on segments that are adapted to the analyzed piece and that are expressed in relationship with the tempo period. We evaluate and analyze our results on a new database composed of classical music pieces.

Contents

6.1 Introduction

In the previous chapter, we have focused on two important musical attributes, the chord progression and the downbeats. The elements of the melody and the harmony of a musical fragment are related to each other by the musical key. Because the musical key is deeply related to the chord progression and the metrical structure, their automatic estimation should be improved by exploiting their interrelationship. In this chapter, we are interested in understanding how the musical key may interact with the musical attributes that we have considered in the previous chapters.

In the first part of this chapter, we focus on the problem of global key estimation, that is finding the main key of a piece of music. We extend the model presented in the previous chapter by integrating global key information and considering interaction between chords, downbeats and the musical key. Relying on previous works on key estimation [Pee06b] [LS08], we study the influence of these three musical attributes on each other. Note that we do not present a new technique for simultaneous estimation of key and chords from a HMM. However, we do present an original analysis about the interaction of musical attributes that shows how they can be estimated in a mutually informing manner.

In the second part of this chapter, we draw our attention to the problem of local key estimation. Various approaches have been proposed in previous works for estimating the global key of a piece of music. Finding the main key of a piece of music is only one part of tonality analysis. Indeed, even if a piece of music generally starts and ends in a particular key referred to as the main or global key of the piece, it is common that the composer moves between keys, sometimes without definitely establish them. A change in the musical key is called a *modulation*. The problem of local key estimation is quite more complex: we aim at segmenting the music piece according to the key changes and finding the key of each segment. Little work has been conducted on this topic. We propose to address the problem of local key finding by investigating the possible combination and extension of different previous proposed global key estimation approaches introducing key dependency on the harmonic and the metrical structures. We show that our model for simultaneous estimation of chords and meter structure can be used to detect the key progression of a music audio file.

Although the idea to use chords to find the key of a musical excerpt has already been explored [NS07], to our knowledge, no precise study about the relationship between the two attributes has been conducted, in particular in the case of local key estimation. This partly comes from a lack of databases labeled in chords and local keys. One contribution of this work is to present such a study on classical music pieces labeled in chords and keys containing many modulations. The problem of finding a good analysis window length for local key estimation has been evoked in the past, without any satisfying answer. Another contribution of our work is that we address this problem by introducing information related to the metrical structure in our model. Key estimation is not performed on empirical-chosen segment length but on segments that are adapted to each piece.

The major contributions presented in this chapter are listed below.

1. We provide a detailed review of previous works about musical key estimation from audio including the problem of local key estimation and the problem of interaction between key and other musical attributes.

2. We integrate global key information in our previous chords/downbeats HMM relying on previous works on key finding.

3. We improve the method proposed in Chapter 4 for training the chord transition matrix and underline the importance of taking into account key information.

4. We propose an analysis of the interaction between the various considered musical attributes using 55 Beatles songs for which we have annotated the key

5. We investigate the possible combination and extension of previously proposed methods for global key estimation to the case of local key estimation.

6. We study the problem of finding an appropriate analysis window length for local key estimation and address this problem by introducing key dependency on the meter.

7. We annotated a novel set of classical music in chords, local key, beats and downbeats.

8. We carefully study the relationship between chords, metrical structure and local key relying on experimental results obtained on the above-mentioned classical music database.

6.1.1 Organization of the chapter:

The structure of the chapter is as follows. First, in Section 6.2, we review some previous works on global and local key estimation. We then investigate the problem of integrating global key information in our model in Section 6.3. In Section 6.3.4, the model is evaluated on a set of popular music pieces. We present in Section 6.4 our model for local key estimation, which relies on the previously proposed probabilistic model for simultaneous chord progression and downbeat locations estimation. The local key estimation is based on the harmonic and metrical structures of the piece. In Section 6.4.5, the proposed model is evaluated on a set classical music pieces. A conclusions section closes the chapter.

6.2 Related work

In this section, we review some previous works related to the problem of the automatic estimation of musical key of a piece of music. We distinguish between works that address the problem of finding the main key (global key) and works that address the problem of key modulations in music pieces.

6.2.1 Global Key

Most of the algorithms that extract key form audio start by computing a set of features that represent the signal (typically chroma features or Pitch Class Profile features), which are then used as an input to a tonality induction model. The problem of automatically estimating the key of a piece of music has first been addressed in the context of symbolic music (*e.g.* MIDI format). In what follows, we will review two of the most popular techniques that have been extended later to the case of audio music. For a detailed review of key finding in symbolic music and more generally on tonality induction, we refer the reader to [Che00] or [GÓ6a].

6.2.1.1 Template-Based Key-finding Models (Krumhansl & Schmuckler Algorithm)

A large part of audio global key finding systems is based on the use of key profiles/templates. Pitch Class Profiles of Chroma features are extracted from the signal and then compared to theoretical templates that represent the perceptual importance of notes or chords within a key. This idea was first proposed by Krumhansl and Schmuckler in [Kru90]. This algorithm, known as the Krumhansl & Schmuckler (K-S) algorithm, computes the correlation between a vector of pitch-class durations obtained from a musical passage and a set of major and (harmonic) minor key-profiles corresponding to each key. The key profile that provides the maximum correlation is taken as the most probable key of the musical excerpt.

The key profiles, known as the Krumhansl-Kessler (K-K) profiles, were originally proposed by Krumhansl and Kessler in [KK82]. They represent the stability of each semitone pitch-class relative to each key. Relying on the idea that some tones are more central than others in a given tonal context, the probe-tone experiments aim at quantifying the hierarchy of stability of the tones within a musical context. The so-called Krumhansl's probe tone ratings were obtained by asking subjects to listen to a musical excerpt that establish a particular key (such as a cadence) and to rate how well each of the 12 semitones of the chromatic scale "fit" the given tonal context. Temperley proposed a modified version of the K-K profiles in [Tem99]. For instance, he adjusted the weights of some pitches in order to give the major and minor profile the same mean. This removed the inherent preference for the minor profile. More details can be found in [Tem01]. He also proposed some modifications to the K-S algorithm by ignoring note durations by using a flat input/weighted key profile approach and by using a matching formula instead of correlation. The Krumhansl and Temperley major and minor key profiles are represented in Figures 6.1 and 6.2.

Various template-based key-finding approaches that rely on the Krumhansl & Schmuckler's approach have been proposed. They differ from each other in either the way audio features are extracted, key templates are chosen or final selection criteria are used.

Gómez & Herrera [GH04] compare a cognitive-based approach with several machine

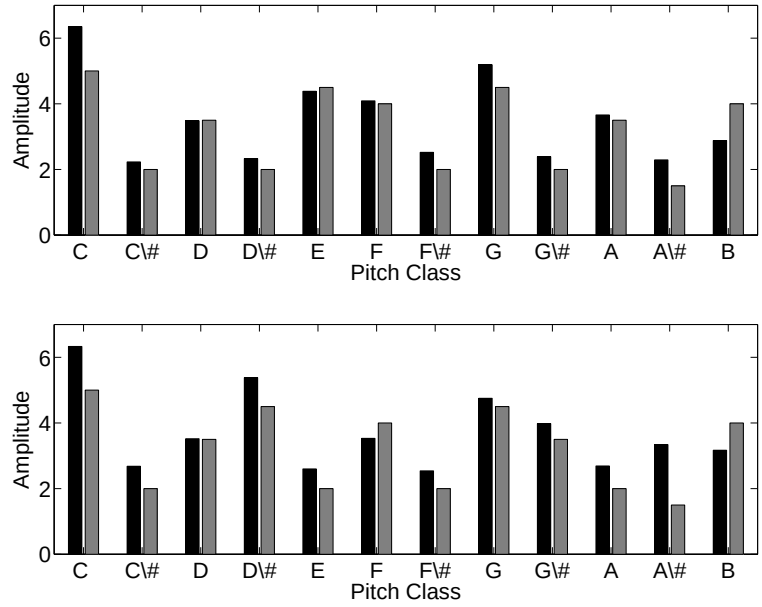

Figure 6.1: Krumhansl (in black) and Temperley (in grey) key profiles for major [top] and minor [bottom] keys.

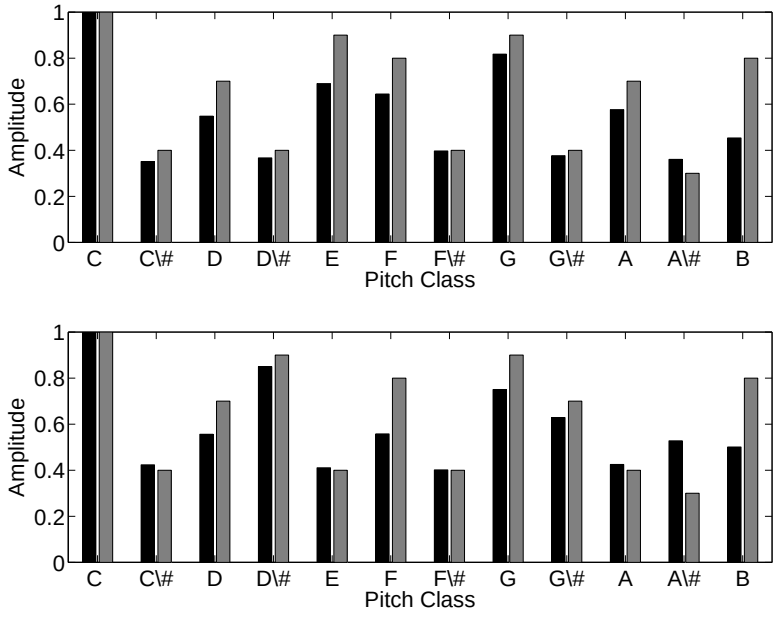

Figure 6.2: Krumhansl (in black) and Temperley (in grey) key profiles for major [top] and minor [bottom] keys normalized by their maximum value.

learning approaches for computing the tonality from polyphonic audio files. A set of Harmonic Pitch Class Profile (HPCP) features is extracted from the signal. In the cognitive-based method, they are used as an input to the Krumhansl & Schmuckler's algorithm that is extended to the case of polyphonic audio by considering the profile value for a given pitch class to represent also the hierarchy of a chord in a given key. The polyphonic profiles for the 24 different keys are built considering only the three main triads of the keys (tonic, subdominant and dominant). This cognition-inspired method is compared with several machine-learning techniques where the key is modeled by analyzing a training annotated collection. Three sub-problems are addressed : mode induction, key note induction and combined key note and mode induction. The methodologies are evaluated over a large audio database, achieving a 64% of correct overall tonality (mode and key-note) estimation. The results on 878 excerpts of classical music (661 for training and 217 for testing) show that combining the two approaches allows improving the key estimation results since both generate different error patterns. It is also found that the use of machine-learning algorithms result in very little improvements over the cognitive-based technique. It is underlined that the considered tonal descriptors do not capture any musical structure, as for instance the position of the chords in the rhythmic or the harmonic structure and that this point needs further investigations.

Pauws [Pau04] also uses the maximum-key profile correlation algorithm to extract the key from the audio signal. For this he correlates chroma features averaged over variable-length fragments at various locations in the piece (beginning, middle, end) with the 24 major/minor Krumhansl and Kessler key-profile vectors. Evaluation on 237 pieces of classical piano sonatas shows that the position and the length of the segments chosen to estimate the main key of the piece is an important parameter. Firstly, a sufficiently long segment is needed to estimate the key. Secondly, the beginning of the piece gives the best information about the key whereas the middle corresponds to a modulation in general and thus leads to an incorrect key estimation. The best results are obtained by analyzing the complete audio file. Some weaknesses of the algorithm are outlined: it does not take into account some parameters that are important for the perception of key such as the position of the pitches in a metrical organization or theoretical and compositional music rules.

Izmirli [Izm05] introduces a template-based correlational model for key finding from audio and compares two methods that implement this model. The first one uses a pure spectral representation and the second uses a chroma-based representation. Key templates are created from monophonic piano audio samples weighted by profiles representing tonal hierarchies in Western music. Three different profiles and their combination are used in this study to approximate the pitch distributions: the Krumhansl's probe tone ratings [Kru90], the Temperley's profiles [Tem01] and a flat diatonic profile (12-dimensional vector containing 1 at pith classes that are comprised in the considered diatonic scale, 0 elsewhere). For each analyzed piece, a spectral or a chroma summary vector is calculated in a similar way than the templates. The key is obtained by computing the correlation between the spectral summary information obtained from audio input and the pre-computed templates. Evaluation is conducted on 85 classical music pieces. The best results are obtained using Temperley's profile combined with a flat diatonic profile.

Izmirli improves this method in [Izm06] where he proposes to use a confidence weighted

correlation to find the most probable key. Although the method is related to the one presented in [GH04] and [GÓ6b], the criteria for key selection is different. In [GÓ6b], the key is estimated by selecting the key-profiles that has the highest correlation coefficient with a global averaged chroma vector. In [Izm06], a summary chroma vector is obtained by averaging the chroma vectors in a window of fixed length, considering different lengths. All windows start from the beginning of the piece and their length is comprised between one frame and 30 seconds. For each window, a key is chosen according to the key profile that has the highest correlation with the summary chroma vector. The global key is then estimated from the individual keys determined on the various window sizes. The model is evaluated using an audio collection consisting of 152 classical pieces.

If most of the template-based key-finding approaches are based on chroma features, some others use different input features. Zhu & Kankanhalli [ZKG05] [ZK06] investigate the use of novel pitch profile features for key detection in musical audio based on the standard probe tone rating method [Kru90]. The novel pitch profiles features address the issues of pitch mistuning and interference of the noisy percussive sounds in the audio. The note partials are precisely extracted from the spectrum using a tuning detection algorithm. A consonance filtering technique is used to select only the note partials that are likely related to the tonality of the music so that peaks corresponding to noisy percussive sounds are filtered out. The advantage of the newly proposed pitch profile feature over the chroma feature is that it is insensitive to temporal variations or dynamics variations in the signal. The proposed system is evaluated on classical and popular music (60 pop songs and 12 classical pieces in [ZKG05], 64 pop songs and 185 15s to 30s-length excerpts of classical music [ZK06]). It is reported that the proposed system performs better using the newly proposed pitch profile features than using the chroma features.

Some works propose to use trained key profiles from either symbolic or audio data instead of using pre-defined key-profiles. This idea was already proposed in [Kru90] where pitch classes distributions were obtained from the melodic line of classical music pieces by counting the number of times that each note appears. It was found that these distributions were closely correlated with the K-K profiles. Temperley proposes in [Tem05] to derive major and minor key profile from the Kostka and Payne corpus [KP95]. Purwins & Blankertz [PB05] use constant-Q profiles trained on audio.

The method proposed in [Pau04] is extended by van de Par *et al.* in [vdPMR06] who propose a similar template-based key-finding approach using trained key-profiles instead of pre-defined key-profiles. For this, frame-by-frame chromagrams are extracted from training audio files and averaged across the duration of the song considering three different weighted functions. The three chromagrams are correlated with the key profiles, yielding to three correlation values with various temporal emphasis (uniform weighting, emphasis on the beginning of the song, emphasis on the ending of the song). The key is determined by computing a weighted score based on the three values. The method is evaluated on the same database as in [Pau04].

Most of the previously presented template-based approaches for key-finding follow the K-S algorithm and are based on the computation of the correlation between pitch class distributions and key-templates. This method does not take into account the temporal

order of the notes although this information may be useful for key-finding. Relying on this observation, Madsen & Widmer [MW07] propose to use distributions of intervals, which is an extension of the pitch class profiles that takes into account the order of the notes. The interval profiles are learned from annotated MIDI data. The key of a given piece is determined by first computing a count table from its pitches and then computing the correlation of this table with each of the major and minor trained profiles. The performance of the model is evaluated on The Finnish Folk Song Database[1] and 384 chorales and 30 inventions by J. S. Bach. The obtained results favorably compare against methods relying on pitch class profiles. It is concluded that the results could be improved by combining the two methods.

6.2.1.2 Key-finding Models Based on HMMs

HMM-based methods have also been proposed as an alternative to template-based methods for key finding. Peeters [Pee06a] investigates the different processes on which template-based key finding approaches rely on. The conclusions of these investigations are presented through the results of key estimation on a database of 302 baroque, classical and romantic music tracks. A Harmonic Peak Subtraction algorithm is proposed as a front-end for the spectral representation of the signal that allows reducing the influence of the higher harmonics of each pitch. Various combinations of key-chroma profiles and key decision methods inspired from [Izm05] and [GÓ6b] are tested. The cognitive-based approach is compared with a HMM-based method for key estimation proposed in [Pee06b].

In the work of Peeters [Pee06b], two hidden Markov models are trained using the Baum-Welsh algorithm on a labeled database in order to learn the characteristics of the major and minor modes. For this, the chroma-vectors of the whole training set are all mapped to a root-note of C and used to train the CM and the Cm key models. The 24 hidden Markov models corresponding to the various major and minor keys are then derived from the two previously trained models by circular permutation. The key of the audio file is obtained by computing the likelihood of its chroma sequence given each HMM and selecting the one that gives the highest value. Note that, in this work, the states in the HMMs have no musical meaning. It is found that the cognitive-based method outperforms the HMM-based approach. It is underlined that the HMM results however show that a system without any a priori musical knowledge can learn the characteristics of the keys from a labeled database. It is mentioned that the results strongly depend on the music genre. Note that these results are similar to the conclusions obtained in [GH04] where it was found that very small improvement is achieved by only using machine learning algorithms. The use of HMM-based methods for key finding has been extended in other works that consider more complex tasks such as the problem of local key finding or the problem of interaction between the key and other musical attributes. We will review these works in the next section.

[1] Available: http://www.jyu.fi/musica/sks/

6.2.1.3 Key-finding Models Based on the Spiral Array Model

Although the K-S model is perhaps the most popular key-finding model, the Spiral Array Model proposed by Chew [Che00] [Che01], originally designed for symbolic key-finding, has also successfully been extended to the case of audio. It is a 3-dimensional model that represents pitches, intervals, chords and keys in the same three-dimensional space. This model is illustrated in Figure 6.3. It can be seen that pitches are represented as points on a helix, adjacent pitches are related by intervals of perfect fifths and vertical neighbors are related by major thirds. The tonal objects are represented as a weighted sum of their lower level components by the center of effect (CE). For instance, the right part of Figure 6.4 represents the CE of the pitches E, F and A weighted by their pitch strengths. Each chord and key is thus given a characteristic point in the space. The key of a musical passage can be estimated using the Center of Effect Generator (CEG) algorithm [Che02]. Pitch and duration information is used to generate the CE of all tonal events in the space. It is compared with the different keys in the spiral array. The closest one is selected as the key of the musical excerpt.

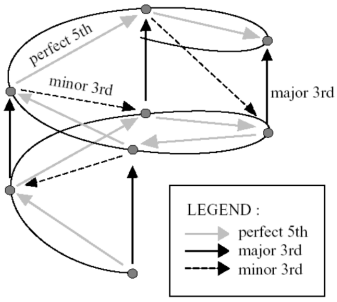

Figure 6.3: Representation of intervals in the Spiral Array model. Adapted from [Che00] and [G06a].

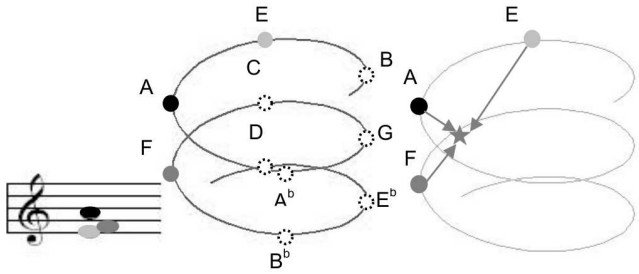

Figure 6.4: Mapping of the pitch strengths of notes E, F and A onto Spiral Array model [middle] and corresponding Center of Effect (CE) [right]. Adapted from [CC05a].

Chuan & Chew [CC05b] propose a key-finding algorithm from polyphonic audio music by extending the Spiral Array CEG algorithm to the case of audio. Pitch class and pitch strength information is extracted from a standard FFT based on a heuristic peak selection algorithm. This information is used as an input of three algorithms for key estimation that are compared: the CEG algorithm, the Krumhansl & Schmuckler's probe tone profile

method and the Temperley's modified version of the K-S method. The performances of the algorithms are compared through a set of 61 audio files of Mozart symphonies synthesized from MIDI. The results show that the CEG algorithm performs better than the Krumhansl and the Temperley methods.

The weak points of this method have been investigated and improved later: Chuan & Chew [CC05a] examine several sources of errors in pitch class determination for audio key finding and propose a fuzzy analysis method to reduce errors in pitch class determination. This method enables to correct noisy detection of lower pitches and to refine the biased raw frequency data. The key is determined by applying the CEG algorithm. The newly proposed method is compared with the previous one that uses peak detection [CC05b] and to a symbolic key finding method. The three algorithms are compared on audio files synthesized from MIDI. Results on the first 15 seconds of a larger and more varied corpus of music than the previous one, composed of 410 pieces of classical music ranging from the Baroque period to the Contemporary period, show that the fuzzy analysis technique performs better than a simple peak detection policy and gives results that are close to the one obtained using symbolic information for pitch detection.

6.2.2 Local Key

In comparison to the problem of global key estimation, little work has been conducted on the problem of local key estimation. However, various approaches have already been proposed for this task.

The above-mentioned Spiral Array model [Che02] is a geometric tonality model that incorporates simultaneously pitch, interval, chord and key relations. It enables determining points of modulation (key changes) in a piece of music in the symbolic domain. This method is extended to the audio case by Chuan & Chew in [CC]. In this work, a basic system that generates pitch-class information using a fuzzy analysis and calculates key results using the CEG algorithm is introduced. Three key determination policies are investigated (nearest-neighbor (NN), relative distance (RD), and average distance (AD)). Experiments are conducted on 410 classical music pieces by various composers across different time and stylistic periods (from Baroque to Contemporary). It is found that the AD policy gives the best main key estimation results (79%). Three extensions to the basic key finding system are then proposed (the modified spiral array (mSA), fundamental frequency ($f0$) identification, and post-weight balancing (PWB)) and evaluated on Chopin's 24 *Préludes*. Quantitative evaluation of main key estimation is proposed. The problem of local key finding is only considered on some examples.

Another geometric tonality model describing relationship between keys has recently been proposed in [GMGB07]. It is derived from the cognition-based model proposed in [KK82]. Tones are organized so that tonal symmetries within Western tonal music become apparent.

Some approaches rely on a frame-by-frame analysis or use a sliding analysis window.

Purwins *et al.* [PBO00] present an approach to derive an appropriate representation of tone centers based on the audio signal using constant-Q profiles. The constant-Q profiles

are 12-dimensional vectors where each component refers to a pitch class. They are derived from sampled cadential chord progressions and small pieces of music. Tonal centers of a music piece are tracked by computing CQ-profiles of the piece and matching every given CQ-profile with a profile of the reference set using a fuzzy distance. The performances of the model are demonstrated over a Chopin's *Préludes* (op.28 no 20), with profiles trained on the 24 Chopin *Préludes*.

Zhu & Kankanhalli [ZK04] present an approach for detecting multiple keys and locating the key boundaries in the melody of popular songs in MIDI format. Overlapping segments are first extracted form the melody using a diatonic scale model, each one corresponding to a single mode. A modality (key style) analysis then determines the center mode of the melody of each segment. Segments of unrelated modes are eliminated. Key labels and boundaries are determined by grouping the remaining segments. The effectiveness of the method is qualitatively measured by analyzing the output of the model while listening to 50 popular songs including English, Japanese and Chinese songs. The ground truth is unknown but it is claimed that the change of the keys can well be perceived.

In order to study the instantaneous evolution of the tonality of a piece of music, Gómez and Bonnada [GB05] present a tool to visualize the tonal content of polyphonic audio signals. The tonal content of an audio file of music is represented by the instantaneous evolution of the tonality and its strength. The tool enables in particular to measure the effect of the length of the sliding window used for key tracking.

Harte *et al.* [HSG06] propose a method for detecting changes in the harmonic content of musical audio signals. A new model for equal tempered tonal space is introduced. Segmentation of audio signal and preprocessing stage for chord recognition and harmonic classification algorithms using HMMs are the main potential applications.

In some other approaches, the segmentation stage (segmentation of the analyzed piece into segment that correspond to a unique key) is more elaborated.

Temperley [Tem05] proposes a Bayesian key-finding model. The analyzed piece is divided into short segments. The model then searches for the most probable "key structure", where a key structure is a labeling of each segment with a key. Each segment can be expressed as a series of pitch-class sets. The fit of a key of each segment to the pitch-classes in the segment is measured using "key-profiles" derived from the Kostka and Payne corpus [KP95]. The model searches for the most probable key structure using dynamic programming, favoring minimum key changes between segments.

Chai & Vercoe [CV05] propose a HMM-based method to segment musical signals according to the key changes and to identify the key of each segment. The front-end of the system is based on the calculation of a chromagram. The key detection task is divided into two steps: first the key root is estimated without considering the mode because diatonic scales are assumed and relative modes share the same diatonic scale. The mode (major or minor) is then estimated. Classical piano music is employed to test the performances of the proposed method using three measures: recall, precision and label accuracy.

Izmirli [Izm07] proposes an interesting new model for detecting modulations and labeling local keys using a non-negative matrix factorization method for segmentation. To

identify sections that are candidates for unique local keys, groups of contiguous chroma vectors are used as input in the segmentation stage. The length of the window is chosen empirically. The local keys are then found using a correlation model. The method is evaluated on three different data sets: pop songs, classical music and Kosta and Payne corpus [KP95].

6.2.3 Key Estimation Methods Based on Chord Progression

In the last few years, there has been an increasing interest in modeling high-level information with low-level signal features in the context of music analysis. Because chords and keys are musical attributes closely related to each other in Western tonal music, the idea to use the chord progression of a piece of music to find the keys comes out naturally. As in the case of simultaneous estimation of chords and downbeats, two paths have been explored for simultaneous estimation of chords and keys.

On the one hand, hierarchical frameworks based on rule-based approaches have been proposed. For instance, Shenoy *et al.* [SMW04] present a rule-based approach for determining the key of acoustic musical signals from the chord progression. The succession of chords is estimated from beat-synchronous chroma features, on which symbolic inference is applied. Only major and minor chords are considered. The chords, detected across all the frames, are then populated into a single 24-dimensional vector. For each key, a reference 24-dimensional reference vector that corresponds to the theoretical distribution of major and minor chords within the considered key is constructed. For instance, the major and minor chords that can be constructed around the CM scale using the notes of this scale are respectively CM, FM, GM and Dm, Em, Am. The pattern that returns the highest rank is selected as the one being the key of the song. It is found that analysis over 16 bars (64 beats) of audio is sufficient to determine the key of the song. The results obtained on a set of 20 popular English songs spanning 4 decades of music lead to a key estimation accuracy of 90%. However, chord recognition accuracy is not sufficient to provide usable chord transcription.

On the other hand, statistical frameworks have been proposed. Raphael & Stoddard [RS03] [RS04] present an approach to functional harmonic analysis based on pitch and rhythm relying on symbolic data. A MIDI representation of a music composition is partitioned into sequences of one-measure length. The goal of this work is to associate a label composed of three variables to each period: the tonic (*e.g.* C, C#) and the mode (major or minor) that give the musical key, and the chord characterized by its harmonic function (scale degree, *e.g.* tonic, dominant). The functional analysis of the chord progression is supposed to guide the choice of the key when it is ambiguous. The analysis is performed with a hidden Markov model that allows the simultaneous estimation of chord and key. The success of the model is demonstrated over some examples but a quantitative evaluation is not presented.

In the framework of global key estimation, several HMM-based works that estimate the chords and keys have been proposed. Lee & Slaney [KS07] [LS08] propose key-dependent chord HMMs trained on synthesized audio for chord recognition and global

key estimation. In these approaches, 24 key-dependent HMMs, one for each major and minor keys are built. Key estimation and chord recognition are performed simultaneously selecting the model whose likelihood is the highest. It is observed that the proposed method is similar to [Pee06b] but, whereas in [Pee06b] the states in the HMMs have no musical meanings, in [KS07], hidden states are treated as chords, which also allows identifying the chord sequence.

Some works that address simultaneously the problem of chords and local key using HMMs have also been proposed. As seen in Chapter 4, Burgoyne & Saul [BS05] present a HMM-based model that tracks key simultaneously with chords.

A recent work by Catteau *et al.* [CML07] proposes a probabilistic framework for simultaneously estimating keys and chords. Novel observation likelihood models and chord/key transition models are proposed that are derived from the music theory of Lerdahl [LJ83]. Parameter tuning and system evaluation is performed using four databases: some cadences and modulation, a set of 10 polyphonic audio fragments of a duration of 60s and a set of 96 MIDI-to-wave synthesized fragments: from the MIREX 2005 key detection contest.

Noland & Sandler [NM06] present a HMM technique for estimating the predominant key in a symbolic musical excerpt. The hidden states are the 24 major and minor keys and the observations are pairs of consecutive chords. Human expectation of harmonic relationships is encoded in the model using results from perceptual tests. The parameters of the HMM are trained using hand-annotated chord symbols. This work is extended to the audio case in [NS07]. Although this model has only been evaluated on the case of global key estimation, it could be used for local key estimation. Note that in this case, there is no complete interaction between chords and keys since the key information is not used to estimate the chords.

6.2.4 Summary of the Works on Key Estimation

The various works on audio global and local key estimation are summarized in Tables 6.1 and 6.2 respectively.

6.3 Interaction between Chords, Meter and Global Key

In this part, we are interested in understanding how the musical key may interact with the chord progression and the downbeat locations. For this, we rely on two previous works proposed for global key estimation from audio that we have presented above: key-dependent chord HMMs proposed in [KS07] inspired from [Pee06b]. In these works, 24 key-dependent HMMs, one for each major and minor keys are built. Key estimation and/or chord estimation are performed by selecting the model whose likelihood is the highest. We propose two modifications of these works. Firstly, we define a specific training approach of the key-specific chord transition matrices that is based on counting the chord transitions from score transcriptions. Secondly, we propose a simple post-processing step that allows correcting some key estimation errors that occur frequently.

Table 6.1: Summary of works on global fey finding.

Reference	Approach	Input Features	Computation Key selection criteria	Evaluation Material
Gómez & Herrera [GH04]	K-S algorithm (correlation)	Harmonic Pitch Class Profile	highest correlation with (K-K) profiles extended to polyphonic audio	878 excerpts of classical music (64%)
Pauws [Pau04]	K-S algorithm (correlation)	chroma	maximum-key profile correlation with (K-K) profiles	237 pieces of classical piano sonatas (75.1%)
Shenoy *et al.* [SMW04]	correlation with templates	chord sequence	maximum correlation with theoretical distribution of major and minor chords within a key	20 popular English songs (90%)
Chuan & Chew [CC05a]	CEG algorithm	fuzzy analysis technique for pitch class determination	Nearest key (Euclidian distance) at stopping point	410 classical music pieces (ranging from Baroque to Contemporary) (75.25%)
Izmirli [Izm05]	K-S algorithm (correlation)	spectral and chroma representation	correlation between spectral summary information and weighted KS-T templates	85 classical music pieces (86%)
Purwins & Blankertz [PB05]	correlation with templates	CQT PCP	maximum correlation of long-term profiles (15s) with trained profiles	MIREX 2005 test-set
van de Par *et al.* [vdPMR06]	K-S algorithm (correlation)	chroma	maximum-key profile correlation with trained key profiles with various temporal emphasis	237 pieces of classical piano sonatas (98%)
Peeters [Pee06b]	HMM	chroma	highest likelihood of the chroma sequence given each trained 24 HMM corresponding to each key	302 European baroque, classical and romantic music extracts (81%)
Izmirli [Izm06]	K-S algorithm (confidence weighted correlation)	spectral and chroma representation	correlation between summary chroma vector and weighted KS-T templates	152 classical pieces (85.5%)
Zhu & Kankanhalli [ZK06]	K-S algorithm (Pearson correlation)	Precise Pitch Profiles	maximum-key profile correlation with (K-K) profiles	64 pop songs and 185 15s to 30s excerpts of classical music (53% for classical pieces and 83% for popular pieces)
Lee & Slaney [KS07] [LS08]	HMM	tonal centroid	selection of one key-dependent HMM for chord estimation among 24	• Classic: Bach Prelude in CM and Haydn string quartet Op.3, No.5, measures 1-46 • 2 Beatles albums, *Please Please Me* and *Beatles For Sale* (97%.)
Madsen & Widmer [MW07]	correlation with interval profiles	interval profiles (count table from pitches)	maximum-key profile correlation with trained interval profiles	The Finnish Folk Song Database + 384 chorales and 30 inventions by J. S. Bach (around 80% (78.7% for Folk song))

6.3.1 Overview of the Model

We start from the "double-states" HMM presented in the previous chapter for simultaneous chords/downbeats estimation. As in [KS07], musical key information is taken into account by using a trained transition matrix (TM) specific to each key. The main key of an analyzed piece of music is found by selecting one chord transition matrix among several trained transition matrices that are described in the next part.

Table 6.2: Summary of works on local fey finding.

Reference	Approach	Input Features	Computation	Evaluation Material
Purwins *et al.* [PBO00]	frame-by-frame correlation with profiles	CQ-profiles	closest fuzzy distance between input profile and trained profile	training on the 24 Chopin *Préludes*, evaluation on Chopin's *Préludes* op.28 no 20
Zhu & Kankanhalli [ZK04]	Melody modality analysis	MIDI melody	segmentation of the melody using a diatonic scale model	qualitative evaluation 50 popular songs
Burgoyne & Saul [BS05]	HMM	PCP	• simultaneous chords and keys • Dirichlet distribution unsupervisedly trained with EM • Simplified rules of tonal harmony encoded in the transition matrix	• 5 Mozart symphonies (K. 134, K. 162, K. 181, K.182 and K.183) for training • Mozart Symphony K. 550, Minuet for testing
Chai & Vercoe [CV05]	HMM	chromagram	key detection divided into two steps: root and mode	10 Classical Mozart piano sonatas
Temperley [Tem05]	Bayesian approach	MIDI	• uses key profiles • favors minimum key changes between segments	main key estimation on 1252 excerpts from classical pieces (91.4%)
Catteau *et al.* [CML07]	probabilistic framework		• simultaneous estimation of keys and chords. • chord/key transition models derived from music theory of Lerdahl	10 polyphonic audio fragments (60 seconds) and 96 MIDI-to-wave synthesized fragments
Chuan & Chewin [CC]	CEG algorithm	Pitch-class	• investigates three key-finding algorithms (modified spiral array (mSA), fundamental frequency identification (F0), and post-weight balancing (PWB)) • investigates three key determination policies (nearest-neighbor (NN), relative distance (RD), and average distance (AD))	•410 classical music pieces (ranging from Baroque to Contemporary) • Chopin's 24 *Préludes*
Izmirli [Izm07]	non-negative matrix factorization method for segmentation + correlation with templates	groups of contiguous chroma vectors	key estimation for each segment using [Izm05] correlation model	Kosta and Payne corpus
Noland & Sandler [NS07]	HMM	pairs of consecutive chords	encode human expectation of harmonic relationships	110 Beatles songs (91%)

The flowchart of the investigated system is presented in Figure 6.5. The previous system is represented in the left part of the figure. In this case, the chord estimation is performed using the cognitive-based transition matrix. For the sake of simplicity, we have not represented the 3/4 or 4/4 meter selection stage described in the previous chapter. The right part of Figure 6.5 represents the same model when key information is taken into account by using a trained transition matrix specific to each key.

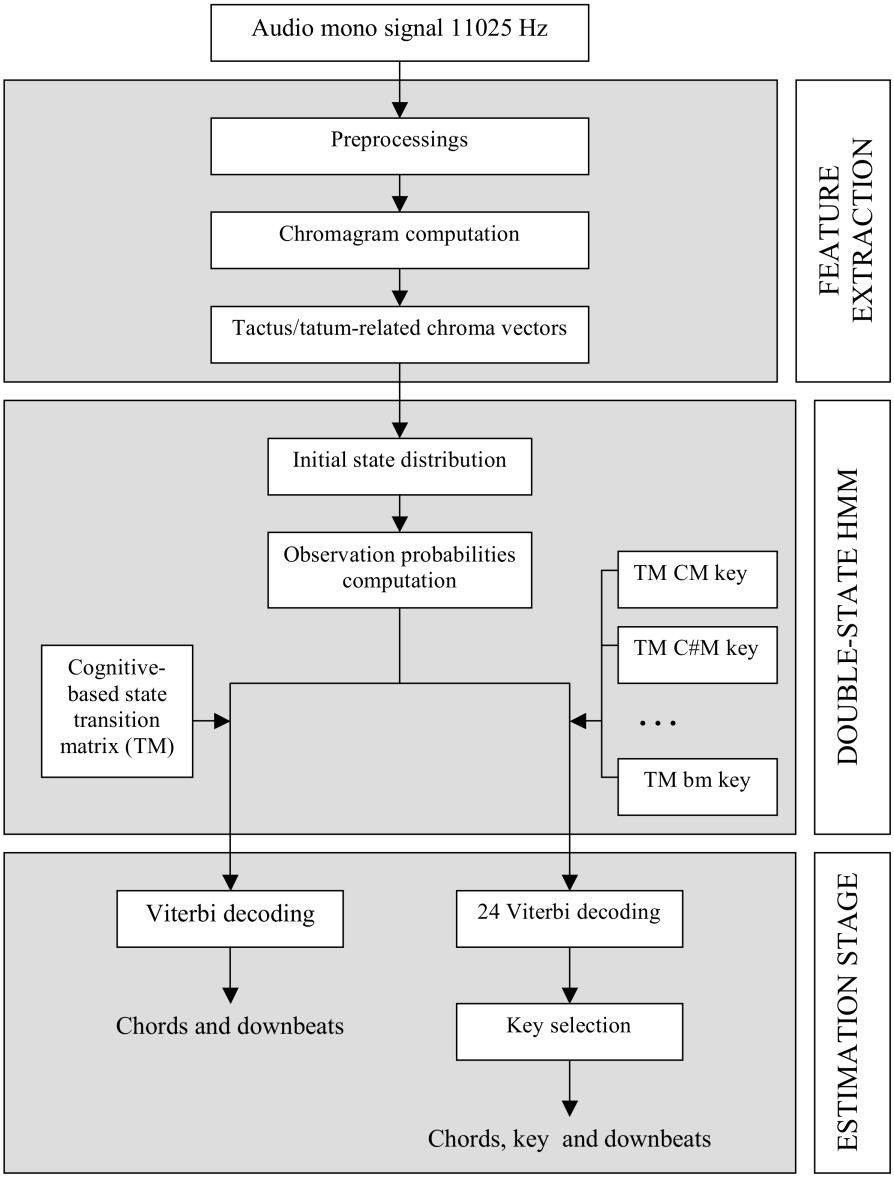

Figure 6.5: General flowchart of the proposed model for simultaneous estimation of the chord progression, the global key and the downbeat locations of an audio file.

6.3.2 Integrating Musical Key Information in the Chord Transition Matrix

We propose here a new method for training the chord transition matrix based on symbolic information, *i.e.* the chord labels transcription of the training set. The probabilities of transiting from one chord to another are learnt by counting the number of occurrences of each chord transition in the training set. This method is very close to the training method of the chord transition matrix, method D, presented in chapter 4, Section 4.4.4.4. The difference is that we introduce here chord transitions dependencies on key in the model.

We build 24 key-specific transition matrices, one for each of the 24 major and minor keys. In order to learn the transition matrices for each of the 24 possible keys, we follow the method proposed in [Pee06b]. We first learn the model for the CM key and the Cm key before mapping the two trained models to all possible keys by circular permutation. The training process is summarized in Figure 6.6 and detailed below:

1. We first assign a global key to each training track (we consider that a training track has a single constant key)[2].
2. We separate the training transcription tracks into two classes according to their mode (major or minor). The two modes are trained separately. We present the training for major mode, the process is exactly the same for minor mode.
3. According to the key of a training track k, we map all its theoretical chords to a reference CM key. This is done by circular permutation of the theoretical chords (transition CM→FM in F major key becomes GM→CM in the reference C major key).
4. For each training segment, we construct a (I,I)-dimensional matrix M_k where $M_k(i,j) = M_k(i,j) + 1$ when the transition from chord i to j occurs in the CM-mapped annotations.
5. The diagonal of the matrix is processed in a separated way and its elements are set to an empirical value. The diagonal values have been empirically set to 0.45 for tatum-frame analysis and 0.4 for tactus-frame analysis in our experiments.
6. All matrices trained on CM-mapped annotations are then averaged and we obtain a single (I,I)-dimensional matrix, denoted by KDC_1 (Key-Dependent Chord matrix).
7. In order to avoid zero-probability chord transitions, we take the exponential of the matrix so that all zero values become non-zero values[3]. The transition matrix is normalized to sum to 1 for each key.
8. The transition matrices for all keys, $KDC_1 \dots KDC_{24}$, are obtained from the two trained models by circular permutation.

Using a test-set composed of 55 Beatles songs described below, we obtain a chord transition matrix KDC_1 for the CM key that is presented in the left part of Figure 6.7. In what follows, we will refer to this matrix as a Key-Dependent Chord (KDC) transition matrix.

We have also represented in the right part of Figure 6.7 the trained transition matrix obtained in Chapter 4 with a similar training method based on counting but ignoring key information (method D). We call this matrix Key-Independent Chord (KIC) transition

[2]The global key assignment is done manually for training.

[3]Note that other choices could have been made to make all the transition matrix values positive. For instance, we could have added 1 to all of the values before normalization.

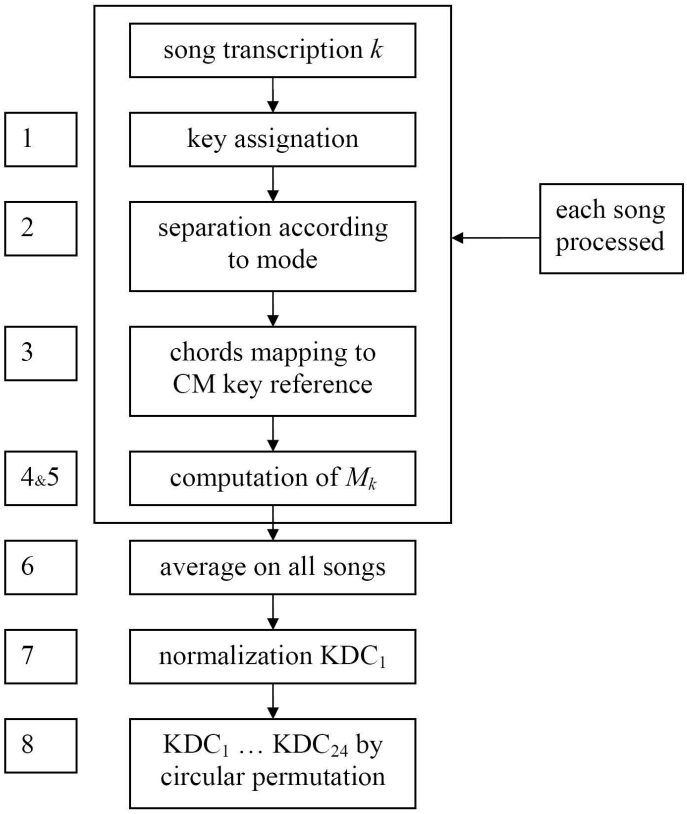

Figure 6.6: Key-dependent chord transition matrices training.

matrix. The training approach allows us to take into account some typical transitions of the test-set. For instance, it can be seen that the transition between a CM chord (predominant chord in general in the key of CM) and a Am chord (relative minor chord of CM) is favored in the KDC matrix, which is not the case in the KIC matrix. It can also be observed that typical transitions in the matrix, such as the II/V/I (transition between Dm, GM and CM) that seems usual in this set of Beatles albums, are predominant in the key-based trained chord transition matrix.

6.3.3 Simultaneous estimation of key, chords and downbeats

6.3.3.1 Key Selection

The system estimates at the same time the key of the track, the chord progression and the downbeats using the Viterbi decoding algorithm [Rab89]. Using successively each trained transition matrix, we obtain simultaneously the best sequence of chords over time and the downbeat positions given the considered key. The musical key of each track is found using a method similar to the one proposed in [KS07]. We select among the 24 possible transitions matrices the one that gives the highest likelihood given the observation sequence $\mathbf{O}$ (*i.e.* the chroma vectors) and the optimal state path $\mathbf{Q}$ (*i.e.* the most probable chord sequence),

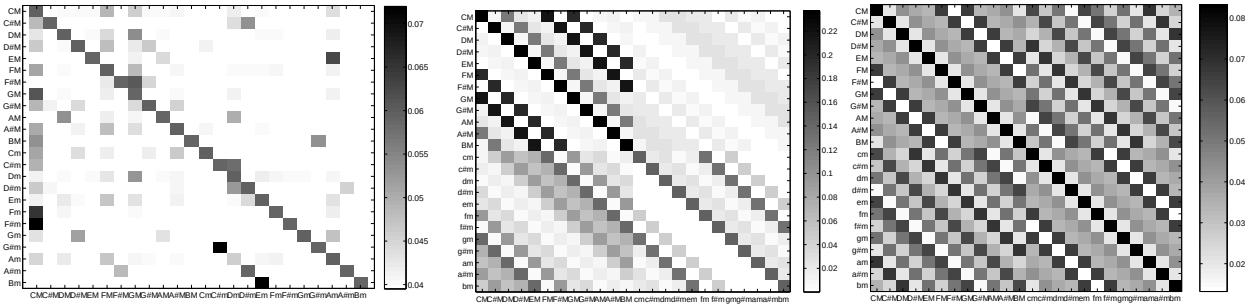

Figure 6.7: From left to right: KDC state transition matrix trained on 55 Beatles songs including key information for a piece in CM key [left]. KIC trained transition matrix obtained without key information based on method D presented in Chapter 4 [middle]. MCC cognitive-based state transition matrix from method B presented in Chapter 4 [right]. Dark marks indicate high values in the transition matrices. Horizontal axis from left to right and vertical axis from top to bottom: chords (CM, C#M, BM, ..., Cm, ..., Bm).

according to Equation (6.1). The estimated key of the track is the one corresponding to the selected transition matrix KDC_i.

$$key = \underset{i}{\operatorname{argmax}}\ P(\mathbf{O}, \mathbf{Q}|KDC_i), i \in [1, 24]. \tag{6.1}$$

6.3.3.2 Post-processing Key Estimation Step

In our experiments, we have found that in many cases, the selected transition matrix corresponds to a key harmonically close to the one of the ground truth. For instance, if the piece is in CM, since G is the dominant of C, it is very likely that many $CM \rightarrow GM$ and $GM \rightarrow CM$ chord transitions will be observed. These transitions are predominant in the transition matrix corresponding to a CM key but also in the one corresponding to a GM key because C is the sub-dominant of G. Therefore, confusions between C major and G major keys are very likely to occur.

To overcome this problem, we propose a post-processing step. We pick up the key transition matrices corresponding to the $K, K \in [1, 24]$ highest likelihood and we compute the proportion of each type of chord detected using each matrix. In practice, K is small and we obtained satisfying results using $K = 4$ in our experiments. It can reasonably be stated that, given a key, the proportion of chords whose root has the same spelling than the considered key is in general the highest. For instance, in a C major piece, there will be a lot of C major chords. We select the transition matrix among the K which gives the chord progression with most of the chords having the same root than the key. In this method, there is an interaction between chords and keys: the key is estimated from the chord transitions and the chord progression is selected among 24 possibilities according to the key.

We thus obtain for each track the chord progression, the downbeats and the main key.

6.3.4 Test-Set and Evaluation Measure

The model has been tested on 55 songs from the first 8 albums of the hand-labeled *Beatles test-set* presented in Chapter 2, Section 2.3.2.

Note that all of the selected songs are in major key. This is because the *Beatles test-set* is very unbalanced in terms of mode: most of the songs have a major main key. Since all the songs have a major global key, we did not have enough songs in minor key for training. We thus evaluate the model considering 12 major key-independent HMMs instead of 24.

Note also that all the selected songs are built on four-beat meter with constant time signature. Moreover, the automatic beat tracking was completely correct for all of them. These characteristics result in a high downbeat detection rate.

For chord evaluation measure, see Section 2.4.2 of Chapter 2. The key estimation evaluation has been performed using a 8-fold cross-validation. We will indicate the rate of correct estimation using two evaluation measures : EE (exact estimation) indicates the percentage of exactly estimated key, ME (MIREX estimation) corresponds MIREX 2005 key estimation measure. For more details, see Chapter 2, Section 2.4.3.1.

6.3.5 Overall Results

The results are indicated in Tables 6.3, 6.4 and 6.5. According to the following list, we investigate the importance of the metrical structure information by taking into account or not the downbeat locations information in the model (With Meter, WM/No Meter, NM). We also investigate the importance of the key information by comparing the results obtained using key-dependent transition matrices (Key-Dependent Chord transition matrix, KDC) with the results obtained using the cognitive-based transition presented in Chapter 4, method B (Music-Cognitive Chord transition matrix, MCC).

Table 6.3: Chords estimation results considering several cases: not integrating/integrating metrical structure information in the model (NM/WM), not integrating/integrating musical key information in the model (MCC/KDC), tactus or tatum-based analysis (TAC/TAT). Stat. Sign. : Statistical significance.

	NM		WM		Stat. Sign.
	TAC	TAT	TAC	TAT	
MCC	74.4 ± 11.7	75.5 ± 12.2	77.5 ± 12.2	78.8 ± 11.3	yes
KDC	75.8 ± 13.3	75.8 ± 13.4	79.3 ± 13.4	78.6 ± 13.5	
Stat. Sign.	yes				

Table 6.4: Key estimation results not integrating/integrating metrical information in the model (NM/WM), tactus or tatum-based analysis (TAC/TAT), exact or MIREX score (EE/ME).

	TAC		TAT	
	EE	ME	EE	ME
NM	83.8	87.5	85.7	92.0
WM	92.9	96.4	89.3	93.8

Table 6.5: Downbeat positions estimation results obtained by the system not integrating/integrating key information in the model (MCC/KDC), tactus or tatum analysis (TAC/TAT).

	TAC	TAT
MCC	98.2	85.7
KDC	96.4	85.7

6.3.6 Analysis of the Results

The results presented in Tables 6.3, 6.4 and 6.5 are briefly summarized in Table 6.6. The downbeat positions and the musical key have been both estimated relying on the chord progression. Conversely, the chord estimation benefits from the knowledge of the key and the downbeat positions. As in the previous chapter, we performed paired sample t-tests at the 5% significance level that has revealed that there is a statistical difference between the chord estimation results obtained with and without considering interaction with the two others musical attributes. In general, taking into account the interaction between the three musical attributes increases their estimation. A detailed analysis follows below.

Table 6.6: Influence of musical context on musical attributes estimation. "Tatum vs. tactus": using a tatum-based analysis rather than a tactus-based analysis.

	Chord	Key	Downbeats
Influence of meter	improvement	improvement	
Influence of key	little improvement		no improvement
Tatum vs. Tactus	improvement	improvement or not	

Importance of the Knowledge of the Downbeat Positions

The results obtained in this chapter corroborate the ones obtained in the previous chapter: integrating chords dependency to the meter allows us to increase the chord recognition rate.

The estimation of the global key of the track is better when taking into account the meter. This comes from the fact that the chord detection accuracy is better.

Importance of the Knowledge of Musical Key Information:

With the proposed model, we obtain up to 92% correctly estimated keys on our evaluation test-set. An analysis of the results shows that most of the errors can be explained and that they correspond to neighboring key confusions (perfect fifth relationship between estimated and ground-truth key). The corresponding MIREX score which takes into account the neighboring keys is up to 96%.

The results in Table 6.3 show that, in most cases, the use of key-specific transition matrices slightly improves the estimation of the chord progression obtained using the cognitive-based matrix (MCC).

It is shown in Table 6.3 that the estimation of the downbeat positions is not improved when taking into account musical key context, probably because the chord estimation is only a little better. Note that the decrease in the results in the case of KDC/tactus-based analysis comes from the fact that for one of the songs, all the downbeats have been estimated on the third beat instead of the first beat of the measures.

6.4 Interaction between Chords, Meter and Local Key

As stated before, the problem of local key finding has been given little attention in the past. We believe that our model can be useful to this task. This is why we now focus on the problem of local key finding in polyphonic audio files. For this, we propose to combine and extend methods proposed for global key finding to the case of local key finding. We rely on the above-mentioned method for global key estimation [GH04] based on key reference profiles, which are correlated with input pitch class profiles. The underlying idea of this work is that in case of polyphonic music, the chords can be used to estimate the musical key. However, in the work of [GH04], as in [Pee06b], there is no estimation of the chords and no investigation of their relationship to keys. We study this relationship in the present work. To integrate the concept of key modulating over time, we propose to use a HMM where the hidden states are the keys which are observed through the chords. The use of the HMM allows us to integrate some musical information about key changes, as proposed in [NM06].

6.4.1 The Problem of the Analysis Window length

As underlined in Section 6.2, HMMs have already been used for local key estimation [NS07], [CV05]. However, this was done using a frame-by-frame analysis. A contribution of the present work is that we introduce information related to the metrical structure of the audio file in order to make the local key estimation robust. One of the problems when segmenting a piece of music into sections with different keys is to accurately choose the length of the analysis window used for key estimation.

In the case of global key estimation, only the first seconds of the piece are used to estimate the key in general. Several studies have shown that the choice of the duration

of the analyzed excerpt has a significant impact on the key estimation results (see for instance [Izm05] or [CC]).

Concerning local key estimation, the length of the analysis window was found empirically in previous works. After computing chroma vectors on short overlapping frames, [CV05] or [PBO00] perform a frame-by-frame musical key analysis. An interesting alternative to sliding window key center tracking techniques has been proposed by [Izm07] where a segmentation stage (whose goal is to identify sections that are candidates for unique local keys) is performed prior to local key estimation. Groups of contiguous chroma vectors are used as input. Heavily overlapped groups of chroma vectors are averaged over a span of σ seconds. The value of the parameter σ is found empirically ($7.4s$) after testing several window sizes.

The question of optimal segment length remains an open problem. A too small window size would focus the chromagram on individual chords more than on keys whereas the use of a too large window size would lead to segments containing several keys and key modulation points would become ambiguous. The drawback of using an empirically chosen window size is that key changes may be ignored by the algorithm for pieces with a fast tempo and that, for pieces with a slow tempo, chords may be estimated rather than keys. Ideally, the window length should be related to the tempo of the piece. We get around this difficulty here by segmenting the piece according to the metrical structure. We perform a beat-synchronous analysis. For local key estimation, the temporal unity, which is used here for key analysis, is the musical bar. The analysis window length has thus a musical meaning.

6.4.2 Model

In this section we present a model that allows estimating the local keys of a musical excerpt using the underlying chord progression, which characterizes the harmonic structure, and the downbeat locations, which characterize the metrical structure. The chords and the downbeats are estimated simultaneously from the sequence of observed chroma vectors using the "double-states" HMM, where a state is a combination of a chord type and a position of the chord in the measure, presented in Chapter 5 of this dissertation. Again, we consider here a chord lexicon composed of the $I = 24$ major and minor triads (CM, ..., BM, Cm, ..., Bm).

The local key estimation model is close to the chord estimation model. Figure 6.8 shows a graph of the HMM we use for local key estimation.

The 24-key space is modeled by an ergodic 24-states HMM, where each state represents one of the 24 major and minor keys. The emission probability of each state (each key) is a 24-dimensional vector representing the probability to observe each of the 24 chords in this specific key. Given the observations, we estimate the most likely key sequence over time in a maximum likelihood sense. The flowchart of the system is represented in Figure 6.9.

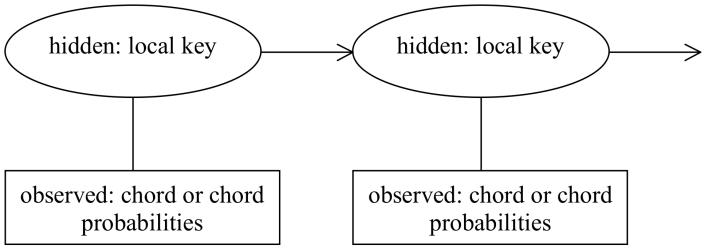

Figure 6.8: Local key estimation hidden Markov model considered in this dissertation. The hidden states correspond to the local keys and the sequence of observations corresponds either to the chord progression or to the instantaneous chord probabilities.

6.4.3 Extraction of Key Observation Vectors

We define the *chordgram* as the succession over time of the 24-dimensional vectors representing the probability to observe each of the 24 chords at each tactus/tatum-frame. These instantaneous chord probabilities correspond to the observation probabilities detailed in the previous chapter (see Section 5.4.4.2, Chapter 5).

In the evaluation part, we will compare two methods for local key estimation. They differ from each other in the way the key observation vectors $\mathbf{O_{key}}$ are derived from the chords.

1. Method 1: The key observation vectors are built from the *chordgram* using the instantaneous chord probabilities $P(\mathbf{O}|c_i)$, where $\mathbf{O}$ corresponds to the chroma vector and $c_i, i \in [1, 24]$ correspond to the chords.

$$\mathbf{O_{key}}(i) = P(\mathbf{O}|c_i) \tag{6.2}$$

2. Method 2: In the second case, the key observation vectors are built directly from the estimated chord progression.

$$\begin{cases} \mathbf{O_{key}}(i) & = 1, \quad \text{if } c_i = \underset{i}{\operatorname{argmax}}\ P(\mathbf{O}|c_i), i \in [1, 24], \\ & = 0 \quad \text{otherwise} \end{cases} \tag{6.3}$$

In general, the musical key of a music piece changes much less often that the chords and remains the same during several bars. We segment the piece into overlapping segments whose length is related to the measures delimited by the downbeats. The local key is thus estimated on segments that have a musical meaning. Because musical phrase have often length duration of 8 or 4+4 bars, we have chosen to segment the pieces into 2-bars segments with 1-bar overlap. Because key changes occur in general on the first beat of a measure it is important that the analysis starts on a downbeat (see Figure 6.12, case OD).

In our experiments we have tested the algorithm using other window analysis length and found that the local key estimation results accuracy decreases with longer windows. This is discussed below in Section 6.4.5.2. The key observation vectors are 24-dimensional vectors obtained by averaging the *chordgram* or the estimated chord progression along the

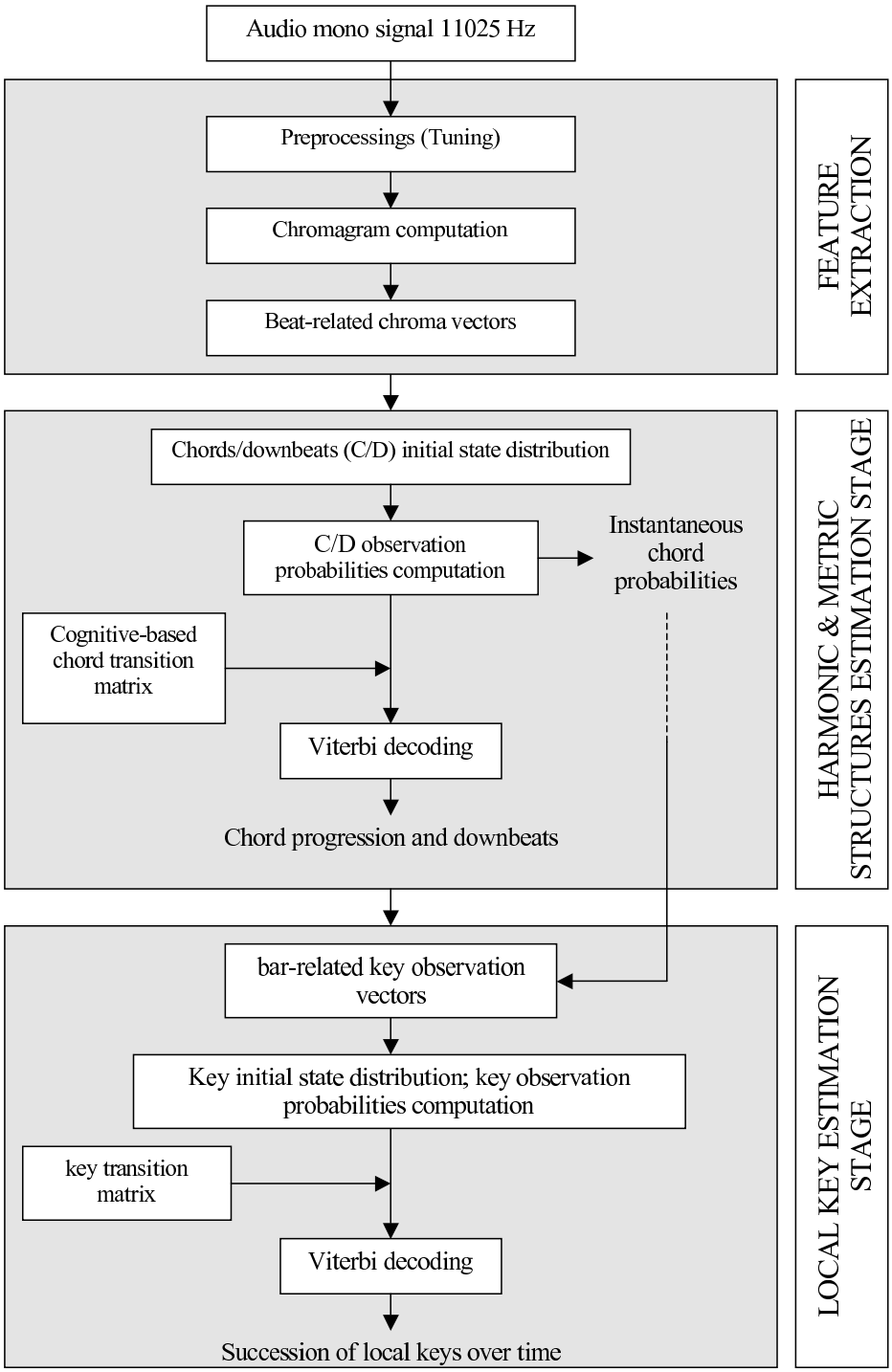

Figure 6.9: Flowchart of the local key estimation system.

overlapping 2-bars length segments. These 24-dimensional vectors represent the probability to observe each of the 24 chords in a specific key.

6.4.4 Key Estimation From Chords Using Hidden Markov Models

From the key observation vectors, we estimate the succession of keys in the track. The method is very similar to the one we proposed for chord estimation.

6.4.4.1 Initial State Distribution

The initial state distribution of keys is uniform ($\frac{1}{24}$ for each of the 24 states) since we have no reason to prefer a key above another.

6.4.4.2 Observation Probabilities of Keys

The observation key probabilities $P(k_i|\mathbf{O_{key}})$ are obtained by computing the cosine distance between the key observation vectors and a set of pre-defined key profiles that represent the importance of each triad within a given key. The pre-defined key templates are 24-dimensional vectors with each bin corresponding to one of the 24 major and minor triads. We have tested our model using four key templates as described below.

The first three of them are derived from the knowledge that the most important triads in a given key are those built on the tonic, the subdominant and the dominant [Kru90] [GH04]. For instance, for a CM key, this chords correspond to CM (C-E-G), FM (F-A-C) and GM (G-B-D).

1. In the first pre-defined key template, we attribute a value of 1 to each of the three main triads. It will be referred to as "main chords" (MC) key template in the following.

2. The second key template is similar to the first one, except that we attribute a higher value $k > 1$ to the chord corresponding to the tonic of the key. In our experiments, we used $k = 3$. It will be referred to as "weighted main chords" (WMC) key template in the following.

3. The third key template is similar to the second one, except that we attribute a value of one to the chord relative to the one built on the tonic (for instance Am chord in a C major key). We consider this case because we have seen that this chord is important in a given key. For instance, the transition CM-Am has a high value in the key-dependent chord transition matrix proposed in the previous section, as it can be seen in Figure 6.7. This key template will be referred to as "weighted main chords relative" (WMCR) in the following.

These three key templates corresponding to the C major (top) and C minor (bottom) keys are represented in Figure 6.10.

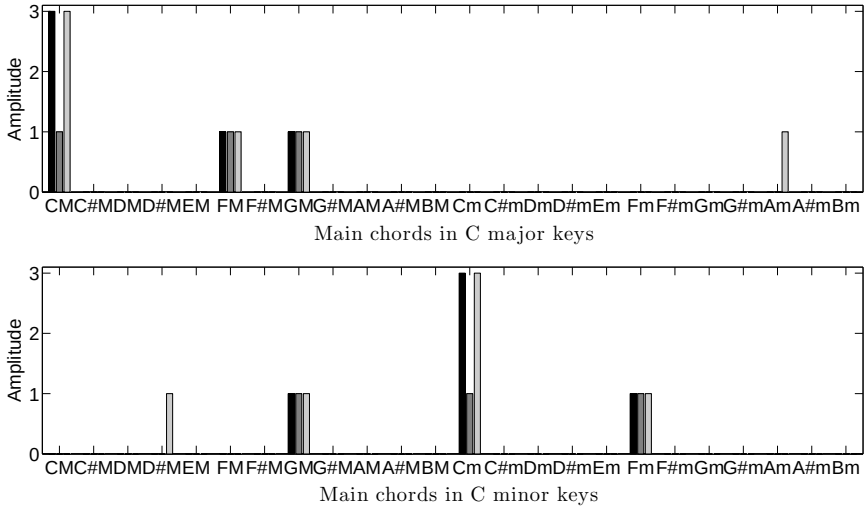

Figure 6.10: Pre-defined 24-dimensional key templates based on the three main triads. Dark grey: "main chords" (MC), black: "weighted main chords" (WMC), light grey: "weighted main chords relative" (WMCR).

The 4^{th} pre-defined key-template is built relying on a cognitive experiment conducted by Krumhansl [Kru90] that gives values corresponding to the rating of chords in harmonic-hierarchy experiments. In this experiment, the perceived relative structural significance of chords in tonal context is measured. For this, several trials consisting of a strong key-defining context followed by a single chord are presented to listeners. The listeners are asked to rate how well the final chord fit with the preceding key-defining context. In the experiments, three types of chords are considered: major, minor and diminished (see Table 6.7). However, since we consider only major and minor chords in our model, the diminished chords were ignored. The cognitive-based key templates corresponding to the C major (top) and C minor (bottom) keys are represented in Figure 6.11.

The templates corresponding to the various major and minor keys are obtained by circular permutation from the one corresponding to the C major and C minor keys.

Let $\mathbf{T}_i, i \in [1, 24]$ denote a key template. The observation key probabilities $P(\mathbf{O_{key}}(t_m)|k_i(t_m))$ are obtained according to Equation (6.4):

$$For \quad i = 1 \dots 24, \quad P(\mathbf{O_{key}}(t_m)|k_i(t_m)) = \frac{\mathbf{O_{key}}(t_m).\mathbf{T}_i}{\|\mathbf{O_{key}}(t_m)\|.\|\mathbf{T}_i\|} \tag{6.4}$$

They are normalized so that:

$$\sum_i P(\mathbf{O_{key}}(t_m)|k_i(t_m)) = 1.$$

Table 6.7: Krumhansl's rating of chords in harmonic hierarchy experiments, [Kru90] p.171.

	Context	
Chord	C Major Key	C Minor Key
C major	6.66	5.30
C#/Db major	4.71	4.11
D major	4.60	3.83
D#/Eb major	4.31	4.14
E major	4.64	3.99
F major	5.59	4.41
F#/Gb major	4.36	3.92
G major	5.33	4.38
G#/Ab major	5.01	4.45
A major	4.64	3.69
A#/Bb major	4.73	4.22
B major	4.67	3.85
C minor	3.75	5.90
C#/Db minor	2.59	3.08
D minor	3.12	3.25
D#/Eb minor	2.18	3.50
E minor	2.76	3.33
F minor	3.19	4.60
F#/Gb minor	2.13	2.98
G minor	2.68	3.48
G#/Ab minor	2.61	3.53
A minor	3.62	3.78
A#/Bb minor	2.56	3.13
B minor	2.76	3.14
C Dim	3.27	3.93
C#/Db Dim	2.70	2.84
D minor	2.59	3.43
D#/Eb Dim	2.79	3.42
E Dim	2.64	3.51
F Dim	2.54	3.41
F#/Gb Dim	3.25	3.91
G Dim	2.58	3.16
G#/Ab Dim	2.36	3.17
A Dim	3.35	4.10
A#/Bb Dim	2.38	3.10
B Dim	2.64	3.18

6.4.4.3 State Transition Probability Distribution

Key modulations in a music piece follow musical rules that can be reflected in the state transition matrix. To integrate musical knowledge in key transition, we adopt the key transition matrix proposed in [NM06] already used as a chord transition matrix (see Chapter 4, transition matrix method B)[4].

[4]Chords and key are musical attributes related to the harmonic structure and can be modeled in a similar way.

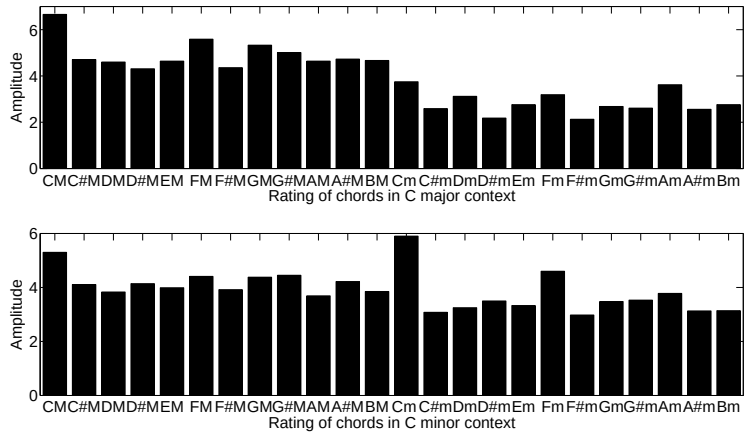

Figure 6.11: Pre-defined 24-dimensional cognitive-based key templates based on Krumhansl's rating of chords in harmonic hierarchy experiments, [Kru90] p.171. Top: C major context. Bottom: C minor context.

6.4.4.4 Local Key Estimation

The optimal succession of states over time is found using the Viterbi decoding algorithm that gives us the best sequence of keys over time. The music piece is thus segmented into segments that are labeled by a key.

6.4.5 Evaluation

6.4.5.1 Test-set and evaluation measures

The proposed model is tested on the *Piano Mozart test-set* presented in Chapter 2. We refer the reader to Section 2.3.3 of this chapter for more details on the *Piano Mozart test-set*. To assess the performance of the system, we use the evaluation measures described in Chapter 2, sections 2.4.

6.4.5.2 Results and discussion

We have carried out several experiments to evaluate the impact of various parameters on the local key estimation results: choice of the key templates, choice of the length of the analysis window, key estimation from the *chordgram* or from the estimated chord progression, influence of the tolerance window.

6.4.5.3 Relationship Between Chords and Local Key

Table 6.8: Chords and local keys label accuracy results using a 2-bars length window and the newly proposed key template WMC. Rex: exact chord estimation rate. Rct: chord estimation rate including close triads. Method 1: based on the chordgram. Method 2: based on the chord progression.

	keys method 1	keys method 2	chords Rex	chords Rct
label accuracy (%)	80.21 ± 13.56	74.11 ± 18.92	61.43 ± 5.50	80.65 ± 8.36

Table 6.9: Local keys segmentation accuracy (SA) results using a 2-bars length window and the WMC proposed templates. Method 1: based on the chordgram. Method 2: based on the chord progression. The tolerance windows is $w = 1$ bar.

	keys method 1	keys method 2
SA precision	0.5723	0.4489
SA recall	0.4730	0.7131
SA f-measure	0.5170	0.5451

We have evaluated the two proposed methods for local key estimation. Recall that:

1. In the first case (method 1), the probability of each chord at a given time instant is used to estimate the key.

2. In the second case (method 2), the chords are first estimated using a hidden Markov model and the local key is derived from the estimated chord progression.

Label and segmentation accuracy results are respectively presented in Tables 6.8 and Table 6.9. Note that we present the results obtained using a window length of 2 measures and using the WMC key templates because we found that these choice of parameters outperformed the others (see below, Sections 6.4.5.5 and 6.4.5.6).

It is difficult to select the best between the two presented methods. Indeed, the best key label results are obtained with method 1, but it can be seen that method 2 slightly outperforms method 2 concerning local key segmentation.

A paired sample t-test at the 5% significance level shows that the difference between the key estimation results obtained with the two methods is not statistically significant. Tests on a larger database would be needed to clearly select the best method.

The analysis of the results piece by piece shows that there is a correlation between the estimation of the chords and the estimation of the key. We expected that a good estimation of the chords would lead to a good estimation of the keys. This was corroborated when evaluating method 2. A good estimation of the chords resulted in a good estimation of the local keys whereas a poor estimation of the chords resulted in a poor estimation of the local keys. A deeper analysis showed that if the chord estimation errors consisted of confusions with harmonically close chords (such as dominant or subdominant chords), the key was nevertheless correctly estimated.

6.4.5.4 Importance of the Metrical Structure

Table 6.10: Local keys results using a 2-bars length window and the WMC templates, when the key analysis windows are set according to the downbeat locations (OD, on downbeats) and when the starting point is not a downbeat (ND, no downbeats). The tolerance window is $w = 1$ bar.

		OD	ND
method 1	label accuracy (%)	80.21	76.43
	segmentation f-measure	0.52	0.50
method 2	label accuracy (%)	74.11	74.80
	segmentation f-measure	0.55	0.52

In Table 6.10, we present the label accuracy results when the key analysis windows are set according to the downbeat locations and when the starting point is not a downbeat. To investigate the hypothesis of the importance of the downbeats on the local key estimation, the key analysis windows have been forced to start on a second beat in case of ND (no downbeats) instead of on a downbeat, as illustrated in Figure 6.12.

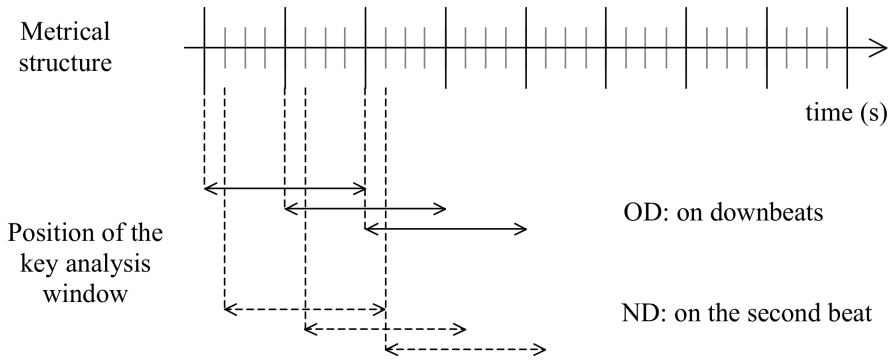

Figure 6.12: Position of the local key analysis window considered for investigating the relationship between the local key and the downbeats. Example for a piece in 4/4 meter.

It can be seen that the label accuracy results are better when the starting point is a downbeat for method 1. This is because key changes occur in general on downbeats. Positioning the starting point of the key analysis window on downbeats helps to avoid mixing some passages with different local keys. Positioning the analysis window on downbeats does not improve the results in the case of method 2. However, the results are not statistically significant.

For both methods, key segmentation results are better when the starting point is a downbeat. However, the difference in the results is slight. This is probably due to the smoothness of the modulations (see below).

Considering the metrical structure seems to improve the key estimation results but tests on a larger database are required to investigate the influence of the metrical structure on the local key estimation.

6.4.5.5 Effect of the Length of the Analysis Window

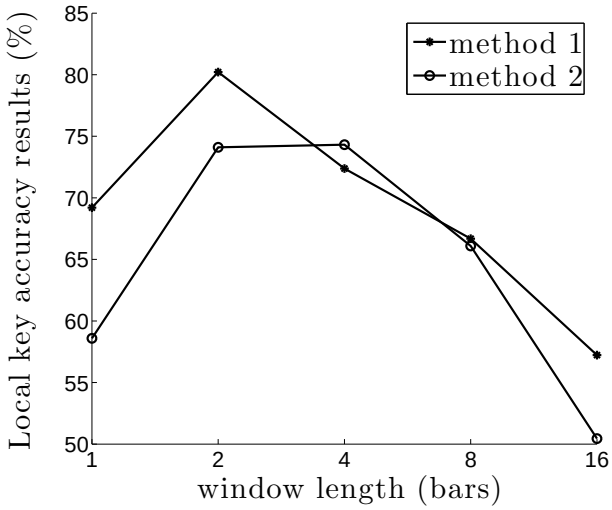

Figure 6.13: Key estimation results in case of method 1 and 2 according to the length of the key analysis window.

In classical music, musical phrases have in general a length of 4 or 8 bars. This is particularly true for Mozart's piano sonatas. Usually, the musical key remains constant within a phrase or at least within half of the phrase (whereas the harmony changes several times). This is why we chose to estimate the local key on segments of length corresponding to musical phrases. We have evaluated the algorithm with different window lengths: 1, 2, 4, 8 and 16 bars. Results are provided in Figure 6.13. A 1-bar analysis window length is too short because it captures the harmony (the chords) rather than the local key. The best results were obtained using a 2-bar length analysis window. This may be due to the fact that, especially in slow movements, some modulations occur after only 2 bars. Passages with different local keys are very likely to be mixed when a longer analysis window is used. The accuracy of the results decreases with the length of the analysis window.

6.4.5.6 Effect of the Choice of the Key Templates

Table 6.11: Local key accuracy results using a 2-bar length window comparing various key templates.

Key template	WMC	MC	WMCR	Krumhansl
Label accuracy (%) meth1	80.21 ± 13.56	79.13 ± 9.01	80.71 ± 11.76	50.02 ± 21.12
Label accuracy (%) meth2	74.11 ± 18.92	71.34 ± 18.42	73.45 ± 19.12	75.69 ± 19.91

We evaluated the algorithm with 4 types of templates. As illustrated in Table 6.11,

the best results are obtained with the weighted main chords WMC templates for method 1. In the case of method 2, the cognitive-based templates slightly outperform the WMC templates. However, statistical tests indicate that in the case of method 2, the difference in the results obtained with the WMC and the cognitive-based templates is note statistically significant. We should perform the evaluation on a larger database to obtain reliable

6.4.5.7 Smooth Modulations:

The key segmentation accuracy results are presented in Table 6.12 in which we consider two tolerance windows: $w = 1$ bar and $w = 2$ bars. It can be seen that the segmentation accuracy results significantly increase when we use a 2-bar tolerance window. This can be explained by the fact that key change is a very smooth process that often takes several bars. It is thus difficult to estimate the precise local key boundaries. It would be interesting to formulate and add a "local key transition" state in the model. This is left for future works.

Table 6.12: Local key segmentation accuracy (SA) results using a 2-bar length window and the proposed WMC templates. Method 1: based on the chordgram. Method 2: based on the chord progression. Two tolerance windows: $w = 1$ bar and $w = 2$ bars.

	keys method 1		keys method 2	
	$w = 1$	$w = 2$	$w = 1$	$w = 2$
SA precision	0.5723	0.8196	0.4489	0.6805
SA recall	0.4730	0.6874	0.7131	0.8691
SA f-measure	0.5170	0.7327	0.5451	0.7514

6.5 Conclusion of the Chapter

In this chapter, we have presented some developments of our model for simultaneous estimation of chords and downbeats by integrating a new musical attribute: the musical key. We have first focused on the problem of global key finding and shown that these three attributes can be estimated in a mutually informing manner that results in some improvements. We have then turned our attention to the problem of local key finding and shown that the local key progression could be estimated relying on the harmonic and the metrical structure of the piece. We have investigated several issues related to this task such as the problem of finding a suitable length of the analysis window or the problem of smooth key changes. Our study has been limited by the little number of annotated pieces we have.

We have presented a local key finding model that segments an audio file in sections labeled with local keys. The method combines and extends several previous methods proposed for global key estimation. The local key progression over time is modeled according to the harmonic and the metrical structures. The local key segmentation has a musical meaning and depends on the tempo of the piece. Encouraging results are obtained on a set of classical pieces with complex harmony structure and show

that the key progression is clearly related to the harmonic and the metrical structures. Analysis of the results shows that additional improvement of key segmentation may be achieved in the future using a more complex model that would include key transition states. Functional chord analysis should also improve the local key estimation [RS04].

f

Chapter 7

Conclusion

Contents

7.1 Thesis Contributions

In this dissertation, we have addressed the problem of estimating content information from music audio signal. The originality of our work is that we estimate simultaneously several musical attributes. Our purpose was to show that a unified music analysis may improve the estimation of individual musical attributes. To that purpose, we have built models that allow the joint estimation of the chords, the keys and the downbeats from polyphonic music recordings. We have demonstrated that integrating knowledge of mutual dependencies between several descriptors of musical content improves their estimation. Part of this PhD work has been devoted to manually annotate some audio databases, which are essential elements of our research.

Building models in which the interaction between musical attributes is encoded at the level musicians and trained human listeners do, when they analyze a piece of music, is a very complex problem and one which is far from being solved. However, we hope that our work is a step towards this direction.

We now summarize the main contributions of the work presented in this thesis.

7.1.1 Features

At the front-end of our models, we extract a *chromagram*, a representation of the signal that captures its harmonic content. We explored several schemes for chromagram computation and investigated several issues related to the use of each representation (problem of harmonics, noise, beat-synchronous features). We conducted a number of experiments on short audio excerpts and proposed some evaluation measures that allow the comparison between the various representations.

The Constant-Q-based chroma features were preferred to the FFT-based ones. They were found to reflect more accurately the harmonic content, especially for popular and rock music that contains lots of percussive sounds: the use of long windows for low frequency allows detecting accurately the bass line, which is very important for chord estimation, whereas the use of short windows for higher frequencies allows reducing the effects of percussive sounds.

Tests on classical piano music showed that the use of multi-f0 features seems to be a promising approach for harmonic content description. However, we did not find this representation convenient for our system since we do not currently have any harmonic/noise separation front-end and thus percussive sounds and noise disrupt the multi-f0s estimation, especially in popular music. Moreover, the rest of our system is computationally very efficient as compared to the multi-f0 analysis. We thus did not favor the use of multi-f0 based chroma features in the rest of our work.

7.1.2 Chords

In our model for joint estimation of musical content descriptors, harmony is considered as a core around which other musical attributes are organized. We thus built a chord

estimation model that serves as a basis for our global model.

We compared several methods based on chroma features and hidden Markov models for the automatic estimation of the chord progression of a music piece. The various methods were compared on a large-scale evaluation on popular music. The best chord estimation results were obtained with the modeling of the observation probabilities using a normalized correlation with a set of extended chord templates and a cognitive-based transition matrix. The templates are extended by considering the presence of higher harmonics of each pitch note of a chord. The transition matrix is derived from cognitive experiments on the perception of musical key.

In our experiments, we found that music knowledge-based models work at least as efficiently as trained models. However, the music knowledge-based transition matrix we propose can only be used for a chord lexicon reduced to the 24 major and minor triads. Probabilistic learning seems to be a solution to extrapolate the proposed model to a larger dictionary. We believe that probabilistic learning could still be exploited and yield to even higher results.

However, since we currently consider a chord lexicon of only 24 triads, we used the HMM-based approach relying on chord templates. This approach gives satisfactory results without requiring any training data. Since this approach does not require training, it allows us to work on various styles of music with the same model.

We used the transition matrix based on Krumhansl's key profiles because we found that this matrix, as well as the one based on the circle of fifths, well characterizes harmonic relationships in a large part of classical and popular music.

7.1.3 Downbeat

The chord estimation model was then modified to integrate information on the metrical structure. We proposed a specific topology of HMM that allowed us to extract simultaneously the chord progression and the downbeats from an audio file.

To that purpose, we first extracted a beat-synchronous chromagram so that the observations capture the harmonic content and are related to the metrical structure. We presented a "double-states" HMM where a state is a combination of a chord type and a position of the chord in the measure. Harmonic and metrical structure information are both encoded in the transition matrix. The chord progression and the downbeats are estimated jointly based on the assumption that chords are more likely to change on the beginning of a measure than on other positions. In order to take into account several cases of metrical structure, two different transition matrices are built. Using a Viterbi decoding algorithm, the most appropriate transition matrix is selected (by selecting the model with highest likelihood). We obtain simultaneously the most likely chord progression and downbeat positions.

An important contribution of our work is that we consider pieces with varying time-signatures and imperfect beat tracking. Most of the previous works assume constant tempo and/or time signature. Any omitted beat or change in tempo or time-signature causes errors from which the downbeat extraction model cannot recover. Our model allows us

to consider pieces with complex metrical structures including changes in the meter from 3/4 to 4/4 time-signature but also various exceptional situations such as the insertions of a measure in 1/4 in a 4/4 meter passage. Our model also allows us to handle errors in the beat tracking stage such as beat insertion or beat deletion due in general to tempo deviation (*e.g.* music tempo speed up or slow down) not detected by the beat tracker.

The system was evaluated and compared to the state-of-the-art on a large set of hand-labeled files. On the one hand, we evaluated its upper limits by estimating the downbeat positions using manual annotation of beat positions. On the other hand, we measured its fully automatic performances by using a beat tracker as a front end. The semi-automatic evaluation has assessed the validity of our model. The fully-automatic evaluation has shown that our model can be applied to real situations in which the beat positions are unknown. By doing so, we have considered the problem of using imperfect beat tracking. Results showed that using a tatum-synchronous analysis instead of a tactus-synchronous analysis might temper the effects of imperfect beat tracking on downbeat tracking.

Comparison with a state-of-the-art downbeat tracking algorithm [DP06] showed that our system, although not perfect, is fairly successful in estimating the downbeats of pieces with complex metrical structure. We demonstrated that considering the interaction between the chord progression and the downbeats allows their simultaneous estimation. We also showed that the chord label accuracy and the chord segmentation accuracy are both improved when estimated jointly with the downbeats.

7.1.4 Key

We have further extended the model to integrate musical key information. We first focused on the problem of finding the main key of a piece of music. For this, the chords/downbeats model was extended by integrating and extending previous works on key estimation [Pee06b] [LS08]. Key information was introduced in the model by using key-dependent chord transitions matrices. We proposed a specific training approach for the key-dependent matrices, from chord labels. We also proposed a simple post-processing step that allows correcting some typical key estimation errors. Analysis on a set of popular music songs has shown that the three musical attributes can be estimated in a mutually informing manner and that each additional musical attribute improves in general the estimation of the others. This has corroborated the idea that a joint estimation of musical parameters improves their estimation.

We then turned our attention to the problem of local key estimation. We proposed to address this problem (segmenting an audio file in sections labeled with local keys) by investigating the possible combination and extension of different previous proposed approaches for global key estimation. The local key progression over time was modeled according to the harmonic and the metrical structure. We investigated several issues related to this task such as the problem of finding a suitable length of the analysis window or the problem of handling smooth key changes. A contribution of our work is that the local key segmentation is based on musical knowledge and depends on the tempo of the analyzed piece. We have shown that our model for simultaneous estimation of chords and downbeats can be used to estimate the key progression of a music audio file. A contribution

of this work is to present a study of the relationships between chords and local keys on an original hand-labeled database of classical music pieces that contain many modulations.

7.2 Future Works

The work proposed in this thesis is a step towards a unified analysis of music. However, there are many issues that still need to be addressed and many potential areas for improvement. We present here some points that we wish to develop in future works.

Concerning the feature extraction part, our experiments show that a pre-processing step that removes transients and noise should be included in our system. This would allow us to improve the chord accuracy. It would also allow us to investigate properly the use of chroma features based on multi-f0s.

We currently use a first-order HMM to model the chord progression. By doing so, we only take into account transitions between consecutive chords. In music pieces, chord sequences exhibit long-term dependencies that should be taken into account in order to model music complexity more accurately. The analysis of chord errors showed that there are ambiguous situations in which the chord estimation would benefit from the knowledge of the tonal function of the chords in the harmonic progression. Modeling chord sequences using longer dependencies between chords, using for instance probabilistic N-grams, would help characterize the complexity of harmonic progressions in Western tonal music [SVB08].

We currently restrict our chord lexicon to the 24 major and minor triads. We think that the proposed model for joint estimation of musical attributes could be directly extended to a larger chord lexicon. However, we did not perform any experiments to corroborate this claim and we let this point for future works.

The downbeat tracking results for pieces in variable meter are encouraging but further improvements are needed. At the present time, the system is built so that it remains in general in a single predominant meter along the analyzed track (although it adjusts to meter changes by inserting measures of different time-signature). It would be highly desirable that the system shows more flexibility to the meter changes. Future work will concentrate on this point. An interesting direction would be to use an observation *pim* distribution that is not uniform (where *pim* corresponds to the position of a beat inside a measure). A possible solution could be to find a way to learn the *pim* distribution from the chord labels.

An analysis of the results shows that the harmonic structure of a piece is an important clue for determining the downbeat positions. However, it was noticed that in some cases (such as when chords change every two beats), the relationship between chord changes and downbeats is ambiguous. This model would benefit from a more complete functional chord analysis. Combining the present system, which is based on harmony, with a rhythmic pattern approach would probably also allow improvement of the downbeat tracking process.

Concerning the problem of local key estimation, the analysis of the results showed that additional improvement of key segmentation may be achieved in the future using a more complex model that would include key transition parts. Functional chord analysis should also improve local keys estimation [RS04].

It must be noticed that the three musical attributes considered (key, chords, downbeats) do not completely interact in the proposed model for local key estimation. Indeed, if the local key is estimated relying on the chord progression and the metrical structure, the estimation of these two elements does not depend on the local key. We plan to consider this point in future works. We believe that the subject of key estimation and particularly local key estimation deserves more attention and that there is place for a wide range of investigations in this area. However, the prior results that we obtained with our current model may be already useful to some music-content applications such as music mood detection for instance.

At the present time, our models have mainly been tested on popular and classical music. It would be interesting to explore their performances on various other music styles in order to see if our hypothesis can be generalize to a wider range of music types.

In this thesis, we have mainly focused on the interaction between three musical attributes. We wish to extend our approach for automatic music content analysis towards a richer model that would integrate other musical attributes, such as the music structure, the melody and, more generally, any element of musical context that has a place in music understanding.

Annexe A - List of the Beatles songs

List of the tracks and the corresponding albums for the *Beatles test-set*.

Table 7.1: List of the Beatles songs (I).

Album	Number	Title
01 Please Please Me	1	01 I Saw Her Standing There
	2	02 Misery
	3	03 Anna (Go To Him)
	4	04 Chains
	5	05 Boys
	6	06 Ask Me Why
	7	07 Please Please Me
	8	08 Love Me Do
	9	09 P. S. I Love You
	10	10 Baby It s You
	11	11 Do You Want To Know A Secret
	12	12 A Taste Of Honey
	13	13 There s A Place
	14	14 Twist And Shout
02 With The Beatles	15	01 It Won t Be Long
	16	02 All I ve Got To Do
	17	03 All My Loving
	18	04 Don t Bother Me
	19	05 Little Child
	20	06 Till There Was You
	21	07 Please Mister Postman
	22	08 Roll Over Beethoven
	23	09 Hold Me Tight
	24	10 You Really Got A Hold On Me
	25	11 I Wanna Be Your Man
	26	12 Devil In Her Heart
	27	13 Not A Second Time
	28	14 Money
03 A Hard Days Night	29	01 A Hard Day s Night
	30	02 I Should Have Known Better
	31	03 If I Fell
	32	04 I m Happy Just To Dance With You
	33	05 And I Love Her
	34	06 Tell Me Why
	35	07 Can t Buy Me Love
	36	08 Any Time At All
	37	09 I ll Cry Instead
	38	10 Things We Said Today
	39	11 When I Get Home
	40	12 You Can t Do That
	41	13 I ll Be Back
04 Beatles For Sale	42	01 No Reply
	43	02 I m a Loser
	44	03 Baby s In Black
	45	04 Rock and Roll Music
	46	05 I ll Follow the Sun
	47	06 Mr. Moonlight
	48	07 Kansas City- Hey, Hey, Hey, Hey
	49	08 Eight Days a Week
	50	09 Words of Love
	51	10 Honey Don t
	52	11 Every Little Thing
	53	12 I Don t Want to Spoil the Party
	54	13 What You re Doing
	55	14 Everybody s Trying to Be My Baby

Table 7.2: List of the Beatles songs (II).

Album	Number	Title
05 Help	56	01 Help!
	57	02 The Night Before
	58	03 You ve Got To Hide Your Love Away
	59	04 I Need You
	60	05 Another Girl
	61	06 You re Going to Lose That Girl
	62	07 Ticket To Ride
	63	08 Act Naturally
	64	09 It s Only Love
	65	10 You Like Me Too Much
	66	11 Tell Me What You See
	67	12 I ve Just Seen a Face
	68	13 Yesterday
	69	14 Dizzy Miss Lizzie
06 Rubber Soul	70	01 Drive My Car
	71	02 Norwegian Wood (This Bird Has Flown)
	72	03 You Won t See Me
	73	04 Nowhere Man
	74	05 Think For Yourself
	75	06 The Word
	76	07 Michelle
	77	08 What Goes On
	78	09 Girl
	79	10 I m Looking Through You
	80	11 In My Life
	81	12 Wait
	82	13 If I Needed Someone
	83	14 Run For Your Life
07 Revolver	84	01 Taxman
	85	02 Eleanor Rigby
	86	03 I m Only Sleeping
	87	04 Love You To
	88	05 Here, There And Everywhere
	89	06 Yellow Submarine
	90	07 She Said She Said
	91	08 Good Day Sunshine
	92	09 And Your Bird Can Sing
	93	10 For No One
	94	11 Doctor Robert
	95	12 I Want To Tell You
	96	13 Got To Get You Into My Life
	97	14 Tomorrow Never Knows

Table 7.3: List of the Beatles songs (III).

Album	Number	Title
08 Sgt Peppers Lonely Hearts Club Band	98	01 Sgt. Pepper s Lonely Hearts Club Band
	99	02 With A Little Help From My Friends
	100	03 Lucy In The Sky With Diamonds
	101	04 Getting Better
	102	05 Fixing A Hole
	103	06 She s Leaving Home
	104	07 Being For The Benefit Of Mr. Kite!
	105	08 Within You Without You
	106	09 When I m Sixty-Four
	107	10 Lovely Rita
	108	11 Good Morning Good Morning
	109	12 Sgt. Pepper s Lonely Hearts Club Band (Reprise)
	110	13 A Day In The Life
09 Magical Mystery Tour	111	01 Magical Mystery Tour
	112	02 The Fool On The Hill
	113	03 Flying
	114	04 Blue Jay Way
	115	05 Your Mother Should Know
	116	06 I Am The Walrus
	117	07 Hello Goodbye
	118	08 Strawberry Fields Forever
	119	09 Penny Lane
	120	10 Baby You re A Rich Man
	121	11 All You Need Is Love
10 CD1 The Beatles	122	CD1 01 Back in the USSR
	123	CD1 02 Dear Prudence
	124	CD1 03 Glass Onion
	125	CD1 04 Ob-La-Di, Ob-La-Da
	126	CD1 05 Wild Honey Pie
	127	CD1 06 -The Continuing Story of Bungalow Bill
	128	CD1 07 While My Guitar Gently Weeps
	129	CD1 08 Happiness is a Warm Gun
	130	CD1 09 Martha My Dear
	131	CD1 10 I m So Tired
	132	CD1 11 Black Bird
	133	CD1 12 Piggies
	134	CD1 13 Rocky Raccoon
	135	CD1 14 Don t Pass Me By
	136	CD1 15 Why Don t We Do It In The Road
	137	CD1 16 I Will
	138	CD1 17 Julia

Table 7.4: List of the Beatles songs (IV).

11 CD2 The Beatles	139	CD2 01 Birthday
	140	CD2 02 Yer Blues
	141	CD2 03 Mother Nature s Son
	142	CD2 04 Everybody s Got Something To Hide Except Me and M
	143	CD2 05 Sexy Sadie
	144	CD2 06 Helter Skelter
	145	CD2 07 Long Long Long
	146	CD2 08 Revolution 1
	147	CD2 09 Honey Pie
	148	CD2 10 Savoy Truffle
	149	CD2 11 Cry Baby Cry
	150	CD2 12 Revolution 9
	151	CD2 13 Good Night
12 Abbey Road	152	01 Come Together
	153	02 Something
	154	03 Maxwell s Silver Hammer
	155	04 Oh! Darling
	156	05 Octopus s Garden
	157	06 I Want You
	158	07 Here Comes The Sun
	159	08 Because
	160	09 You Never Give Me Your Money
	161	10 Sun King
	162	11 Mean Mr Mustard
	163	12 Polythene Pam
	164	13 She Came In Through The Bathroom Window
	165	14 Golden Slumbers
	166	15 Carry That Weight
	167	16 The End
	168	17 Her Majesty
13 Let It Be	169	01 Two of Us
	170	02 Dig a Pony
	171	03 Across the Universe
	172	04 I Me Mine
	173	05 Dig It
	174	06 Let It Be
	175	07 Maggie Mae
	176	08 I ve Got A Feeling
	177	09 One After 909
	178	10 The Long and Winding Road
	179	11 For You Blue
	180	12 Get Back

Annexe B - List of publications

List of publications by the author related to this thesis:

- H. Papadopoulos and G. Peeters: *Joint Estimation of Chords and Downbeats From Au Audio Signal*. To appear in IEEE Transactions on Audio, Speech, and Language Processing.
- H. Papadopoulos and G. Peeters: *Local Key Estimation Based on Harmonic and Metric Structures*. In Dafx 2009.
- H. Papadopoulos and G. Peeters: *Simultaneous Estimation of Chord Progression and Downbeats from an Audio File*. In ICASSP 2008.
- H. Papadopoulos and G. Peeters: *Large-Scale Study of Chord Estimation Algorithms Based on Chroma Representation and HMM*. In CBMI 2007.

Bibliography

[All04] H. Allan. Bar lines and beyond - Meter tracking in digital audio. Master's thesis, School of Informatics, University of Edinburgh, Edinburgh, UK, 2004.

[AS04] M. Abe and J.O. Smith. Design criteria for simple sinusoidal parameter estimation based on quadratic interpolation of FFT magnitude peaks. In *Proceedings of the Convention Audio Engineering Society (AES)*, San Francisco, CA, USA, October 28-31 2004.

[Bel07] J.P. Bello. Audio-based cover song retrieval using approximate chord sequences: testing shifts, gaps, swaps and beats. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 23-27 2007.

[BP92] J.C. Brown and M.S. Puckette. An efficient algorithm for the calculation of a constant Q transform. *Journal of the Acoustical Society of America*, 92(5):2698–2701, Nov. 1992.

[BP05] J.P. Bello and J. Pickens. A robust mid-level representation for harmonic content in music signal. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 304–311, London, UK, September 11-15 2005.

[BPKF07] J.A. Burgoyne, L. Pugin, C. Kereliuk, and I. Fujinaga. A cross-validated study of modelling strategies for automatic chord recognition in audio. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 251–254, Vienna, Austria, September 23-27 2007.

[Bro91] J.C. Brown. Calculation of a constant Q spectral transform. *Journal of the Acoustical Society of America*, 89(1):425–434, 1991.

[BS05] J.A. Burgoyne and L.K. Saul. Learning harmonic relationships in digital audio with Dirichlet-based hidden Markov models. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, London, UK, September 11-15 2005.

[BW01] M.A. Bartsch and G.H. Wakefield. To catch a chorus using chroma-based representations for audio thumbnailing. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 15–18, New Paltz, NY, USA, October 21-24 2001.

[BW05] M.A. Bartsch and G.H. Wakefield. Audio thumbnailing of popular music using chroma-based representations. *IEEE Transactions on Multimedia*, 7(1):96–104, Feb. 2005.

[CB09] T. Cho and J.P. Bello. Real-time implementation of HMM-based chord estimation in musical audio. In *Proceedings of the International Computer Music Conference (ICMC)*, Montreal, QC, Canada, August 16-21 2009.

[CC] C.-H. Chuan and E. Chew. Audio key finding: considerations in system design and case studies on Chopin's 24 preludes. *EURASIP Journal on Advances in Signal Processing*, 2007:Article ID 56561, 15 pages.

[CC05a] C.-H. Chuan and E. Chew. Fuzzy analysis in pitch class determination for polyphonic audio key finding. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, London, UK, September 11-15 2005.

[CC05b] C.-H. Chuan and E. Chew. Polyphonic audio key finding using the spiral array CEG algorithm. In *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME)*, Amsterdam, Netherlands, July 6-9 2005.

[Che00] E. Chew. *Towards a mathematical model of tonality*. PhD thesis, Department of Operations Research, Massachussets Institute of Technology, Cambridge, MA, USA, 2000.

[Che01] E. Chew. Modeling tonality: applications to music cognition. In *Proceedings of the Annual Meeting of the Cognitive Science Society (COGSCI)*, pages 206–211, Edinburgh, Scotland, August 1-4 2001.

[Che02] E. Chew. The spiral array: an algorithm for determining key boundaries. In *Proceedings of the International Conference on Music and Artificial Intelligence (ICMAI)*, pages 18–31, Edinburgh, Scotland, September 12-14 2002.

[CML07] B. Catteau, J.P. Martens, and M. Leman. A probabilistic framework for audio-based tonal key and chord recognition. In R. Decker and H.-J. Lenz, editors, *Advances in data analysis - Proceedings of the 30th Annual Conference of the Gesellschaft für Klassifikation*, pages 637–644, Berlin, Germany, March 8-10 2007. Springer.

[CT65] J. Cooley and J. Tukey. An algorithm for the machine computation of the complex Fourier series. *Mathematics of Computation*, 19:297–301, Apr. 1965.

[CV05] W. Chai and B. Vercoe. Detection of key change in classical piano music. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, London, UK, September 11-15 2005.

[CYL+08] H.-T. Cheng, Y.-H. Yang, Y.-C. Lin, I.-B. Liao, and H.H Chen. Automatic chord recognition for music classification and retrieval. In *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME)*, Hannover, Germany, June 23-26 2008.

[Dav07] M.E.P. Davies. *Towards automatic rhythmic accompaniment*. PhD thesis, Queen Mary University, London, UK, August 2007.

[DBSD04] C. Duxbury, J.P. Bello, M. Sandler, and M. Davies. A Comparison Between Fixed and Multiresolution Analysis for Onset Detection in Music Signals. In *Proceedings of the International Conference on Digital Audio Effects*, Naples, Italy, October 5-8 2004.

[Dix06] S. Dixon. Onset detection revisited. In *Proceedings of the International Conference on Digital Audio Effects*, pages 133–137, Montreal, QC, Canada, September 18-20 2006.

[DP06] M.E.P Davies and M.D. Plumbley. A spectral difference approach to downbeat extraction in musical audio. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Florence, Italy, September 4-8 2006.

[EA04] D.P.W. Ellis and J. Arroyo. Eigenrhythms: drum pattern basis sets for classification and generation. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Barcelona, Spain, October 10-14 2004.

[ECM08] D.P.W. Ellis, C.V. Cotton, and M.I. Mandel. Cross-correlation of beat-synchronous representations for music similarity. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processings (ICASSP)*, pages 57–60, Las Vegas, NV, USA, March 30-April 4 2008.

[Ell06] D.P.W. Ellis. Identifying 'cover songs' with beat-synchronous chroma features. In *MIREX*, Victoria, BC, Canada, October 8-12 2006.

[Ell07] D.P.W. Ellis. Classifying music audio with timbral and chroma features. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 23-27 2007.

[Ell08] D.P.W. Ellis. Simple trained audio chord recognition. In *MIREX*, Philadelphia, PA, USA, September 14-18 2008.

[EP07] D.P.W. Ellis and G.E. Poliner. Identifying 'cover songs' with chroma features and dynamic programming beat tracking. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processings (ICASSP)*, volume 4, pages 1429–1432, Honolulu, HI, USA, April 15-20 2007.

[Fuj99] T. Fujishima. Real-time chord recognition of musical sound: a system using common lisp music. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 464–467, Beijing, China, October 22-28 1999.

[GÓ6a] E. Gómez. *Tonal description of music audio signals*. PhD thesis, Universitat Pompeu Fabra, Barcelona, Spain, 2006.

[GÓ6b] E. Gómez. Tonal description of polyphonic audio for music content processing. *INFORMS Journal on Computing*, 18(3):294–304, 2006.

[GB05] E. Gómez and J. Bonada. Tonality visualization of polyphonic audio. In *Proceedings of the International Computer Music Conference (ICMC)*, Barcelona, Spain, September 5-9 2005.

[GBC07] M. Gainza, D. Barry, and E. Coyle. Automatic bar line segmentation. In *Proceedings of the Convention Audio Engineering Society (AES)*, New York, NY, USA, October 5-8 2007.

[GH04] E. Gómez and P. Herrera. Estimating the tonality of polyphonic audio files: cognitive versus machine learning modelling strategies. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 92–95, Barcelona, Spain, October 10-14 2004.

[GHNO02] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: popular, classical, and jazz music databases. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 287–288, Paris, France, October 13-17 2002.

[GM94] M. Goto and Y. Muraoka. A Beat tracking system for acoustic signals of music. In *Proceedings of the ACM International Conference on Multimedia*, pages 365–372, San Francisco, CA, USA, October 15-20 1994.

[GM96] M. Goto and Y. Muraoka. Beat tracking based on multiple-agent architecture a real-time beat tracking system for audio signals. In Mario Tokoro, editor, *Proceedings of the International Conference on Multiagent Systems (ICMAS)*, pages 103–110, Kyoto, Japan, December 1996.

[GM99a] B. Gold and N. Morgan. *Speech and audio signal processing: processing and perception of speech and music*. John Wiley & Sons Inc., 1999.

[GM99b] M. Goto and Y. Muraoka. Real time beat tracking for drumless audio signals: chord change detection for musical decisions. *Speech Communication*, 27:311–335, 1999.

[GMAB08] G. Gatzsche, M. Mehnert, D. Arndt, and K. Brandenburg. Circular pitch space based musical tonality analysis. In *Proceedings of the Convention Audio Engineering Society (AES)*, Amsterdam, Netherlands, May 17-20 2008.

[GMGB07] G. Gatzsche, M. Mehnert, D. Gatzsche, and K. Brandenburg. A symmetry based approach for musical tonality analysis. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 23-27 2007.

[Got01] M. Goto. An audio-based real-time beat tracking system for music with or without drum sounds. *Journal of New Music Research*, 30(2):159–171, 2001.

[Got06] M. Goto. A chorus section detection method for musical audio signals and its application to a music listening station. *IEEE Transactions on Audio, Speech and Language Processing*, 14(5):1783–1794, Sep. 2006.

[Hai04] S.W. Hainsworth. *Techniques for the automated analysis of musical audio.* PhD thesis, Department of Engineering, Cambridge University, Cambridge, UK, 2004.

[Har78] F.J. Harris. On the use of windows for harmonic analysis with the discrete fourier transform. *Proceedings of the IEEE*, 66(1), 1978.

[HB08] J.A. Hockman and J.P. Bello. Automated rhythmic transformation of musical audio. In *Proceedings of the International Conference on Digital Audio Effects*, Espoo, Finland, September 1-4 2008.

[HS05] C.A. Harte and M.B. Sandler. Automatic chord identification using a quantised chromagram. In *Proceedings of the Convention Audio Engineering Society (AES)*, Barcelona, Spain, May 28-31 2005.

[HSAG05] C. Harte, M. Sandler, S. Abdallah, and E. Gómez. Symbolic representation of musical chords: a proposed syntax for text annotations. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, London, UK, September 11-15 2005.

[HSG06] C. Harte, M. Sandler, and M. Gasser. Detecting harmonic change in musical audio. In *Proceedings of the Audio and Music Computing for Multimedia Workshop (AMCMM)*, Santa Barbara, CA, USA, October 27 2006.

[Izm05] Ö Izmirli. Template based key finding from audio. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 211–214, Barcelona, Spain, September 5-9 2005.

[Izm06] Ö. Izmirli. Audio key finding using low-dimensional spaces. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 127–132, Victoria, BC, Canada, October 8-12 2006.

[Izm07] Ö Izmirli. Localized key finding from audio using non-negative matrix factorization for segmentation. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 23-27 2007.

[Jeh05] T. Jehan. Downbeat prediction by listening and learning. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, October 16-19 2005.

[KEA06] A.P. Klapuri, A. Eronen, and J. Astola. Analysis of the meter of acoustic musical signals. *IEEE Transactions on Audio, Speech and Language Processing*, 14(1):342–355, 2006.

[KK82] C.L. Krumhansl and E.J. Kessler. Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys. *Psychological Review*, (89):334–368, 1982.

[KM10] N. Konoki, Y. Emura and M. Miura. Chord estimation using chromatic profiles of sounds played by an electric guitar. In *Proceedings of the International Conference on Music Perception & Cognition (ICMPC)*, pages 734–737, Seattle, WA, USA, August 23-27 2010.

[KO09] M. Khadkevich and M. Omologo. Use of hidden Markov models and factored language models for automatic chord recognition. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Kobe, Japan, October 26-30 2009.

[KP95] S. Kostka and D. Payne. *Tonal harmony.* McGraw Hill, New York, NY, USA, 1995.

[Kru90] C.L. Krumhansl. *Cognitive foundations of musical pitch.* Oxford University Press, New York, NY, USA, 1990.

[KS07] Lee. K. and M. Slaney. A unified system for chord transcription and key extraction using hidden Markov models. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 23-27 2007.

[KWP06] Y.E. Kim, D.S. Williamson, and S. Pilli. Towards quantifying the ‚Äúalbum effect‚Äù in artist identification. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 393–394, Victoria, BC, Canada, October 8-12 2006.

[LB07] E. Li and J.P. Bello. Key-independent classification of harmonic change in musical audio. In *Proceedings of the Convention Audio Engineering Society (AES)*, New York, NY, USA, October 5-8 2007.

[Lee05] K. Lee. Automatic chord recognition using a summary autocorrelation function. Technical report, Stanford EE391, Spring 2005.

[Lee06a] K. Lee. Automatic chord recognition from audio using enhanced pitch class profile. In *Proceedings of the International Computer Music Conference (ICMC)*, New Orleans, LA, USA, November 6-11 2006.

[Lee06b] K. Lee. Identifying cover songs from audio using harmonic representation. In *MIREX*, Victoria, BC, Canada, October 8-12 2006.

[Lee08] K. Lee. *A system for acoustic chord transcription and key extraction from audio using hidden Markov models trained on synthesized audio.* PhD thesis, Stanford University, CA, USA, March 2008.

[LJ83] F. Lerdahl and R. Jackendoff. *A Generative Theory of Tonal Music.* MIT press, 1983.

[LS08] K. Lee and M. Slaney. Acoustic chord transcription and key extraction from audio using key-dependent HMMs trained on synthesized audio. *IEEE Transactions on Audio, Speech and Language Processing*, 16(2):291–301, Feb. 2008.

[Mad06] N.C. Maddage. Automatic structure detection for popular music. *IEEE MultiMedia*, 13(1):65–77, 2006.

[MD08] M. Mauch and S. Dixon. A discrete mixture model for chord labelling. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Philadelphia, PA, USA, September 14-18 2008.

[MD10] M. Mauch and S. Dixon. Simultaneous estimation of chords and musical context from audio. *IEEE Transactions on Audio, Speech and Language Processing*, 2010. To appear.

[MDH+07] M. Mauch, S. Dixon, C. Harte, M. Casey, and B. Fields. Discovering chord idioms through Beatles and real book songs. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 23-27 2007.

[MEK09] M. Müller, S. Ewert, and S. Kreuzer. Making chroma features more robust to timbre changes. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processings (ICASSP)*, pages 1877–1880, Taipei, Taiwan, April 19-24 2009.

[MGAZ08] M. Mehnert, G. Gatzsche, D. Arndt, and T. Zhao. Circular pitch space based chord analysis. In *MIREX*, Philadelphia, PA, USA, September 14-18 2008.

[MKC05] M. Müller, F. Kurth, and M. Clausen. Chroma-based statistical audio features for audio matching. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 275–278, New Paltz, NY, USA, October 16-19 2005.

[MKL06] N.C. Maddage, M.S. Kankanhalli, and H. Li. A hierarchical approach for music chord modeling based on the analysis of tonal characterics. In *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME)*, Toronto, ON, Canada, July 9-12 2006.

[MND09] M. Mauch, K. Noland, and S. Dixon. Using musical structure to enhance automatic chord transcription. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Kobe, Japan, October 26-30 2009.

[MW07] S.T. Madsen and G. Widmer. Key-finding with interval profiles. In *Proceedings of the International Computer Music Conference (ICMC)*, Copenhagen, Denmark, August 27-31 2007.

[MXKS04] N.C. Maddage, C. Xu, M.S. Kankanhalli, and X. Shao. Content-based music structure analysis with applications to music semantics understanding. In *Proceedings of the ACM International Conference on Multimedia*, New York, NY, USA, October 10-16 2004.

[NM06] K. Noland and Sandler M. Key estimation using a hidden Markov model. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 121–126, Victoria, BC, Canada, October 8-12 2006.

[NS07] K. Noland and M. Sandler. Signal processing parameters for tonality estimation. In *Proceedings of the Convention Audio Engineering Society (AES)*, Vienna, Austria, May 5-8 2007.

[OGF09a] L. Oudre, Y. Grenier, and C. Févotte. Chord recognition using measures of fit, chord templates and filtering methods. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, October 18-21 2009.

[OGF09b] L. Oudre, Y. Grenier, and C. Févotte. Chord recognition by fitting rescaled chroma vectors to chord templates. Technical report, Telecom Paritech, Octobre 2009.

[Pau04] S. Pauws. Musical key extraction from audio. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 96–99, Barcelona, Spain, October 10-14 2004.

[PB05] H. Purwins and B. Blankertz. CQ-profiles for key finding in audio. In *MIREX*, London, UK, September 11-15 2005.

[PBO00] H. Purwins, B. Blankertz, and K. Obermayer. A new method for tracking modulations in tonal music in audio data format. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, Como, Italy, July 24-27 2000.

[PBO01] H. Purwins, B. Blankertz, and K. Obermayer. Constant Q profiles for tracking modulations in audio data. In *Proceedings of the International Computer Music Conference (ICMC)*, La Habana, Cuba, September 17-22 2001.

[PC02] J. Pickens and T. Crawford. Harmonic models for polyphonic music retrieval. In *Proceedings of the International Congress of Mathematicians (ICM)*, Beijing, China, August 20-28 2002.

[PEBB05] J.-F. Paiement, D. Eck, S. Bengio, and D. Barber. A graphical model for chord progressions embedded in a psychoacoustic space. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 641–648, Bonn, Germany, August 7-11 2005.

[Pee] G. Peeters. Template-based estimation of time-varying tempo. *EURASIP Journal on Advances in Signal Processing*, 2007:Article ID 67215, 14 pages.

[Pee06a] G. Peeters. Chroma-based estimation of tonality from audio-signal analysis. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 115–120, Victoria, BC, Canada, October 8-12 2006.

[Pee06b] G. Peeters. Musical key estimation of audio signal based on HMM modeling of chroma vectors. In *Proceedings of the International Conference on Digital Audio Effects*, pages 127–131, Montreal, QC, Canada, September 18-20 2006.

[Pee09] G. Peeters. Beat-marker location using a probabilistic framework and linear discriminant analysis. In *Proceedings of the International Conference on Digital Audio Effects*, Como, Italy, September 1-4 2009.

[PP07] H. Papadopoulos and G. Peeters. Large-Scale Study of Chord Estimation Algorithms Based on Chroma Representation and HMM. In *Proceedings of the International Workshop on Content-Based Multimedia Indexing (CBMI)*, pages 53–60, Bordeaux, France, June 25-27 2007.

[PP08a] H. Papadopoulos and G. Peeters. Chord estimation using chord templates and HMM. In *MIREX*, Philadelphia, PA, USA, September 14-18 2008.

[PP08b] H. Papadopoulos and G. Peeters. Simultaneous estimation of chord progression and downbeats from an audio file. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processings (ICASSP)*, pages 121–124, Las Vegas, NV, USA, March 30-April 4 2008.

[PP09] H. Papadopoulos and G. Peeters. Local key estimation based on harmonic and metric structures. In *Proceedings of the International Conference on Digital Audio Effects*, Como, Italy, September 1-4 2009.

[PP10] H. Papadopoulos and G. Peeters. Joint estimation of chords and downbeats. *IEEE Transactions on Audio, Speech and Language Processing*, 2010. To appear.

[Rab89] L. Rabiner. A tutorial on hidden Markov model and selected applications in speech. *Proceedings of the IEEE*, 77(2):257–285, 1989.

[Rap02] C. Raphael. Automatic transcription of piano music. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Paris, France, October 13-17 2002.

[RK08a] M. Ryynänen and A. Klapuri. Chord detection method for Mirex 2008. In *MIREX*, Philadelphia, PA, USA, September 14-18 2008.

[RK08b] M.P. Ryynänen and A.P. Klapuri. Automatic transcription of melody, bass line, and chords in polyphonic music. *Computer Music Journal*, 32(3):72–86, 2008.

[RS03] C. Raphael and J. Stoddard. Harmonic analysis with probabilistic graphical models. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 177–181, Baltimore, MD, USA, October 26-30 2003.

[RS04] C. Raphael and J. Stoddard. Functional harmonic analysis using probabilistic models. *Computer Music Journal*, 28(3):45–52, 2004.

[RSN08] J. Reinhard, S. Stober, and A. Nürnberger. Enhancing chord classification through neighbourhood histograms. In *Proceedings of the International Workshop on Content-Based Multimedia Indexing (CBMI)*, London, UK, June 18-20 2008.

[SE03] A. Sheh and D.P.W. Ellis. Chord segmentation and recognition using EM-trained HMM. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 183–189, Baltimore, MD, USA, October 26-30 2003.

[Ser07] J. Serrà. Music similarity based on sequences of descriptors: tonal features applied to audio cover song identification. Master's thesis, Department of Information and Communication Technologies, Universitat Pompeu Fabra, Barcelona, Spain, September 2007.

[SGHS08] J. Serrà, E. Gómez, P. Herrera, and X. Serra. Chroma binary similarity and local alignment applied to cover song identification. *IEEE Transactions on Audio, Speech and Language Processing*, 16(6):1138–1151, Aug. 2008.

[SHAR09] B. Schüller, B. Hörnler, D. Arsic, and G. Rigoll. Audio chord labelling by musicological modeling and beat synchronization. In *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME)*, New York, NY, USA, June 28-July 3 2009.

[She88] C. Sher. *The New Real Book*. Sher Music Co., 1988.

[SIY+08] K. Sumi, K. Itoyama, K. Yoshii, K. Komatani, T. Ogata, and H.G Okuno. Automatic chord recognition based on probabilistic integration of chord transition and bass pitch estimation. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Philadelphia, PA, USA, September 14-18 2008.

[Smi08] J.O. Smith. *Spectral audio signal processing*. `http://ccrma.stanford.edu/ jos/sasp/`, October 2008. Online book.

[SMW04] A. Shenoy, R. Mohapatra, and Y. Wang. Key determination of acoustic musical signals. In *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME)*, volume 3, pages 1771–1774, Taipei, Taiwan, June 27-30 2004.

[SP09] A.M. Stark and M.D. Plumbley. Real-time chord recognition for live performance. In *Proceedings of the International Computer Music Conference (ICMC)*, Montreal, QC, Canada, August 16-21 2009.

[SSG+09] M. Stein, B.M. Schubert, M. Gruhne, G. Gatzsche, and M. Mehnert. Evaluation and comparison of audio chroma feature extraction methods. In *Proceedings of the Convention Audio Engineering Society (AES)*, Munich, Germany, May 7-10 2009.

[SVB08] R. Scholz, E. Vincent, and F. Bimbot. Robust modeling of musical chord sequences using probabilistic N-grams. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processings (ICASSP)*, Las Vegas, NV, USA, March 30-April 4 2008.

[SW05] A. Shenoy and Y. Wang. Key, chord and rhythm tracking of popular music recordings. *Computer Music Journal*, 3(29):75–86, 2005.

[Tem99] D. Temperley. What's key for key? The Krumhansl-Schmuckler key-finding algorithm reconsidered. *Music Perception*, 17(1):65–100, 1999.

[Tem01] D. Temperley. *The cognition of basic musical structures*. MIT Press, Cambridge, MA, USA, 2001.

[Tem05] D. Temperley. A Bayesian key-finding model. In *MIREX*, London, UK, September 11-15 2005.

[TK00] T. Tolonen and M Karjalainen. A computationally efficient multipitch analysis model. *IEEE Transactions on Audio, Speech and Language Processing*, 8(6):208–216, 2000.

[TS99] D. Temperley and D. Sleator. Modeling meter and harmony: a preference-rule approach. *Computer Music Journal*, 23(1):10–27, 1999.

[UMOS08] Y. Uchiyama, K. Miyamoto, N. Ono, and S. Sagayama. Automatic chord detection using harmonic sound emphasized chroma from musical acoustic signal. In *MIREX*, Philadelphia, PA, USA, September 14-18 2008.

[vdPMR06] S. van de Par, M.F. McKinney, and A. Redert. Musical key extraction from audio using profile training. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 328–329, Victoria, BC, Canada, October 8-12 2006.

[VPM08] M. Varewyck, J. Pauwels, and J.P. Martens. A novel chroma representation of polyphonic music based on multiple pitch tracking techniques. In *Proceedings of the ACM International Conference on Multimedia*, pages 667–670, Vancouver, BC, Canada, October 27-31 2008.

[Wak99] G.H. Wakefield. Mathematical representation of joint time-chroma distribution. In *Proceedings of the SPIE Conference on Advanced Signal Processing Algorithms, Architecture and Implementation*, pages 637–645, Denver, CO, USA, July 19-21 1999.

[WFS01] B. Whitman, G. Flake, and Lawrence S. Artist detection in music with Minnowmatch. In *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing*, pages 559–568, Falmouth, MA, USA, September 10-12 2001.

[YB78] J.E. Younberg and S.F. Boll. Constant-Q signal analysis and synthesis. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processings (ICASSP)*, pages 375–378, Tulsa, OK, USA, April 10-12 1978.

[Yeh08] C. Yeh. *Multiple fundamental frequency estimation of polyphonic recordings*. PhD thesis, Paris 6 University, Paris, France, 2008.

[YKK+04] T. Yoshioka, T. Kitahara, K. Komatani, T. Ogata, and H.G. Okuno. Automatic chord transcription with concurrent recognition of chord symbols and boundaries. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Barcelona, Spain, October 10-14 2004.

[YRC08] C. Yeh, A. Roebel, and W.-C. Chang. Multiple-f0 estimation for MIREX 2008. In *MIREX*, Philadelphia, PA, USA, September 14-18 2008.

[ZK04] Y. Zhu and M.S. Kankanhalli. Key-based melody segmentation for popular songs. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, Cambridge, UK, August 23-26 2004.

[ZK06] Y. Zhu and M.S. Kankanhalli. Precise pitch profile feature extraction from musical audio for key detection. *IEEE Transactions on Multimedia*, 8(3):575–584, 2006.

[ZKG05] Y. Zhu, M.S. Kankanhalli, and S. Gao. Music key detection for musical audio. In *Proceedings of the International Multimedia Modelling Conference (MMM)*, pages 30–37, Melbourne, VIC, Australia, January 12-14 2005.

[ZR07] V. Zenz and A. Rauber. Automatic chord detection incorporating beat and key detection. In *Proceedings of the International Conference on Signal Processing and Communication Systems (ICSPC)*, Gold Coast, Australia, December 17-19 2007.