MUSIC CONTENT ANALYSIS : KEY, CHORD AND RHYTHM TRACKING IN ACOUSTIC SIGNALS

ARUN SHENOY KOTA

(B.Eng.(Computer Science), Mangalore University, India)

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

NATIONAL UNIVERSITY OF SINGAPORE

2004

Acknowledgments

I am grateful to Dr. Wang Ye for extending an opportunity to pursue audio research and work on various aspects of music analysis, which has led to this dissertation. Through his ideas, support and enthusiastic supervision, he is in many ways directly responsible for much of the direction this work took. He has been the best advisor and teacher I could have wished for and it has been a joy to work with him.

I would like to acknowledge Dr. Terence Sim for his support, in the role of a mentor, during my first term of graduate study and for our numerous technical and music theoretic discussions thereafter. He has also served as my thesis examiner along with Dr Mohan Kankanhalli. I greatly appreciate the valuable comments and suggestions given by them.

Special thanks to Roshni for her contribution to my work through our numerous discussions and constructive arguments. She has also been a great source of practical information, as well as being happy to be the first to hear my outrage or glee at the day's current events.

There are a few special people in the audio community that I must acknowledge due to their importance in my work. It is not practical to list all of those that have contributed, because then I would be reciting names of many that I never met, but whose published work has inspired me.

I would like to thank my family, in particular my mum & dad, my sister and my grandparents whose love and encouragement have always been felt in my life. Finally, a big thank you to all my friends, wherever they are, for all the good times we have shared that has helped me come this far in life . . .

Contents

List of Tables

List of Figures

Summary

We propose a music content analysis framework to determine the musical *key*, *chords* and the hierarchical *rhythm structure* in musical audio signals. Knowledge of the key will enable us to apply a music theoretic analysis to derive the scale and thus the pitch class elements that a piece of music uses, that would be otherwise difficult to determine on account of complexities in polyphonic audio analysis. Chords are the harmonic description of the music and serve to capture much of the essence of the musical piece. The identity of individual notes in the music does not seem to be important. Rather, it is the overall quality conveyed by the combination of notes to form chords. Rhythm is another component that is fundamental to the perception of music. A hierarchical structure like the measure (bar-line) level can provide information more useful for modeling music at a higher level of understanding.

Our rule-based approach uses a combination of top down and bottom up approaches - combining the strength of higher level musical knowledge and low level audio features. To the best of our knowledge this is the first attempt to extract *all* of these three important expressive dimensions of music from real world musical recordings (sampled from CD audio), carefully selected for their variety in artist and time spans. Experimental results illustrate accurate key and rhythm structure determination for 28 out of 30 songs tested with an average chord recognition accuracy of around 80% across the length of the entire musical piece. We do a detailed evaluation of the test results and highlight the limitations of the system. We also demonstrate the applicability of this approach to other aspects of music content analysis and outline steps for further development.

Chapter 1

Introduction

1.1 Motivation

Content based analysis of music is one particular aspect of computational auditory scene analysis, the field that deals with building computer models of higher auditory functions. A computational model that can understand musical audio signals in a human-like fashion has many useful applications. These include:

- Automatic music transcription: This problem deals the transformation of musical audio into a symbolic representation such as MIDI or a musical score which in principle, could then be used to recreate the musical piece [36].

- Music informational retrieval: Interaction with large databases of musical multimedia could be made simpler by annotating audio data with information that is useful for search and retrieval [25].

- Emotion detection in music: Hevner [18] has carried out experiments that substantiated a hypothesis that music inherently carries emotional meaning. Huron [19] has pointed out that since the preeminent functions of music are social and psychological, emotion could serve as a very useful measure for the characterization of music in information retrieval

systems. The relation between musical chords and their influence on the listeners emotion has been demonstrated by Sollberger in [47].

- Structured Audio : The first generation of partly-automated structured-audio coding tools could be built [25]. Structured Audio means transmitting sound by describing it rather than compressing it [24]. Content analysis could be used to partly automate the creation of this description by the automatic extraction of various musical constructs from the audio.

While the general auditory scene analysis is something we would expect most human listeners to have reasonable success at, this is not the case for the automatic analysis of musical content. Even simple human acts of congnition such as tapping the foot to the beat, swaying to the pulse or waving the hands in time with the music are not easily reproduced in a computer program [42].

Over the years, a lot of research has been carried out in the general area of music and audio content processing. These include analysis of pitch, beats, rhythm and dynamics, timbre classification, chords, harmony and melody extraction among others. The landscape of music content processing technologies is discussed in [1].

To contribute towards this research, we propose a novel framework to analyze a musical audio signal (sampled from CD audio) and determine its key, provide usable chord transcriptions and determine the hierarchical rhythm structure across the length of the music.

Though the detection of individual notes would form the lowest level of music analysis, the identity of individual notes in music does not seem to be important. Rather, it is the overall quality conveyed by the combination of notes to form chords [36]. Chords are the harmonic

description of the music and serve to capture much of the essence of the musical piece. Non-expert listeners, hear groups of notes as chords. It can be quite difficult to identify whether or not a particular pitch has been heard in a chord. Analysis of music into notes is also unnecessary for classification of music by genre, identification of musical instruments by their timbre, or segmentation of music into sectional divisions [25].

The key defines the diatonic scale which a piece of music uses. The diatonic scale is a seven note scale and is most familiar as the Major scale or the Minor scale in music. The key can be used to obtain high level information about the musical content of the song that can capture much of the character of the musical piece.

Rhythm is another component that is fundamental to the perception of music. A hierarchical structure like the measure (bar-line) level can provide information more useful for modeling music at a higher level of understanding [17].

Key, chords and rhythm are important expressive dimensions in musical performances. Although expression is necessarily contained in the physical features of the audio signal such as amplitudes, frequencies and onset times, it is better understood when viewed from a higher level of abstraction, that is, in terms of musical constructs [11] like the ones discussed here.

1.2 Related Work

1.2.1 Key Determination

Existing work has been restricted to either the symbolic domain (MIDI and score) [4, 27, 33, 40] or single instrument sounds and simple polyphonic sounds [37]. An attempt to extract the musical scale and thus the key of a melody has been attempted in [53, 54]. This approach is again

however restricted to the MIDI domain [53, 54] and to hummed queries [53]. To our knowledge, the current effort is the first attempt to to identify the key from real-world musical recordings.

1.2.2 Chord Determination

Over the years, considerable work has been done in the detection and recognition of chords. However this has been mostly restricted to single instrument and simple polyphonic sounds [5, 6, 13, 21, 28, 39] or music in the symbolic, rather than that in the audio domain [29, 30, 34, 35, 40].

A statistical approach to perform chord segmentation and recognition on real-world musical recordings that uses the Hidden Markov Models (HMMs) trained using the Expectation-Maximization (EM) algorithm has been demonstrated in [44] by Sheh and Ellis. This work draws on the prior idea of Fujishima [13] who proposed a representation of audio termed "pitch class profiles" (PCPs), in which the Fourier transform intensities are mapped to the twelve semitone classes (chroma). This system assumes that the chord sequence of an entire piece is known beforehand. In this chord recognition system, first the input signal is transformed to the frequency domain. Then it is mapped to the PCP domain by summing and normalizing the pitch chroma intensities, for every time slice. PCP vectors are used as features to build chord models using HMM via EM. Prior to training, a single composite HMM for each song is constructed according to the chord sequence information. During the training, the EM algorithm calculates the mean and variance vector values, and the transition probabilities for each chord HMM. With these parameters defined, the model can now be used to determine a chord labeling for each test song. This is done using the the Viterbi algorithm to either forcibly align or recognize these labels. In forced alignment, observations are aligned to a composed HMM whose transitions are limited to those dictated by a specific chord sequence. In recognition, the HMM is

unconstrained, in that any chord may follow any other, subject only to the markov constraints in the trained transition matrix. Forced alignment always outperforms recognition, since the basic chord sequence is already known in forced alignment which then has to only determine the boundaries, whereas recognition has to determine the chord labels too.

1.2.3 Rhythm Structure Determination

A lot of research in the past has focused on rhythm analysis and the the development of beat-tracking systems. However, most of them did not consider the higher-level beat structure above the quarter note level [10, 11, 16, 41, 42, 50] or were restricted to the symbolic domain rather than working in real-world acoustic environments [2, 7, 8, 38].

In [17], Goto and Muraoka have developed a technique for detecting a hierarchical beat structure in musical audio without drum-sounds using chord change detection for musical decisions. Because it is difficult to detect chord changes when using only a bottom-up frequency analysis, a top-down approach of using the provisional beat times has been used. The provisional beat times are a hypothesis of the quarter-note level and are inferred by an analysis of onset times. In this model, onset times are represented by an onset-time vector whose dimensions correspond to the onset times of different frequency ranges. A beat-prediction stage is used to infer the quarter-note level by using the autocorrelation and cross-correlation of the onset-time vector. The chord change analysis is then performed at the quarter note level and at the eighth note level, by slicing the frequency spectrum into strips at the provisional beat times and at the interpolated eighth note levels. This is followed by an analysis of how much the dominant frequency components included in chord tones and their harmonic overtones change in the frequency spectrum. Musical knowledge of chord change is then applied to detect the higher-level rhythm structure at the half and measure (whole note) levels.

In [15], Goto has developed a hierarchical beat tracking system for musical audio signals with or without drum sounds using drum patterns in addition to onset times and chord changes discussed previously. A drum pattern is represented by the temporal pattern of a bass and snare drum. A drum pattern detector detects the onset times of the bass and snare drums in the signal which are used to create drum patterns and then compared against eight pre-stored drum patterns. Using this information and musical knowledge of drum in addition to musical knowledge of chord changes, the rhythm analysis at the half note level is performed. The drum pattern analysis can be performed only if the musical audio signal contains drums and hence a technique that measures the autocorrelation of the snare drum's onset times is applied. Based on the premise that drum-sounds are noisy, the signal is determined to contain drum sounds only if this autocorrelation value is high enough. Based on the presence or absence of drum sounds, the knowledge of chord changes and/or drum patterns is selectively applied. The highest level of rhythm analysis at the measure level (whole note/ bar) is then performed using only musical knowledge of chord change patterns.

1.3 Contributions of this thesis

We shall now discuss the shortcomings in the existing work discussed in the previous section.

The approach for chord detection used in [44] assumes that the chord sequence for an entire piece is known. This has been obtained for 20 songs by the *Beatles* from a standard book of *Beatles* transcriptions. Thus the training approach limits the technique to be restricted to the detection of known chord progressions. Further, as the training and testing data is restricted to the music of only one artist, it is unclear how this system will perform for other kinds of music.

[15, 17] perform real-time higher level rhythm determination up to the measure level us-

ing chord change analysis without identifying musical notes or chords by name. In both these works, it is mentioned that chord identification in real-world audio signals is generally difficult. Traditionally, musical chord recognition is approached as a combination of polyphonic transcription to identify the individual notes followed by symbolic inference to determine the chord [13]. However in the audio domain, various kinds of noise and overlap of harmonic components of individual notes would make this a difficult task. Further, techniques applied to systems that used as their input MIDI-like representations cannot be directly applied because it is not easy to obtain complete MIDI representations from real-world audio signals.

Thus in this work, we propose an offline-processing, rule-based framework to obtain ***all*** of the following from real-world musical recordings (sampled from commercial CD audio):

1. Musical key - to our knowledge, the first attempt in this direction.
2. Usable chord transcriptions - that overcome all of the problems with [44] highlighted above.
3. Hierarchical rhythm structure across the entire length of the musical piece - where the detection has been performed using actual chord information, as against chord change probabilities used in [15, 17].

1.4 Document Organization

The rest of this document is organized as follows. In Chapter 2 we give a primer on music theoretic concepts and define the terminology used in the the rest of this document. In Chapter 3, we give a brief overview of our system. Chapter 4 discusses the individual components of this system in detail. In Chapter 5 we present the empirical evaluation of our approach. Finally, we discuss our conclusion and highlight the future work in Chapter 6.

Chapter 2

Music Theory Background

2.1 Note

A *note* is a unit of fixed pitch that has been given a name. Pitch refers to the perception of the frequency of a note.

2.2 Octave

An *octave* is the interval between one musical note and another whose pitch is twice its frequency. The human ear tends to hear both notes as being essentially the same. For this reason, notes an octave apart are given the same note name. This is called *octave equivalence*.

2.3 Tonic / Key

The word *tonic* simply refers to the most important note in a piece or section of a piece. Music that follows this principle is called *tonal music*. In the tonal system, all the notes are perceived in relation to one central or stable pitch, the tonic. Music that lacks a tonal center, or in which all

pitches carry equal importance is called *Atonal music*. Tonic is sometimes used interchangeably with *key*. All tonal music is based upon scales. Theoretically, to determine the key from a piece of sheet music, the key signature is used. The key signature is merely a convenience of notation placed on the music staff, containing notation in sharps and flats. Each key is uniquely identified by the number of sharps or flats it contains. An example is shown in Figure 2.1

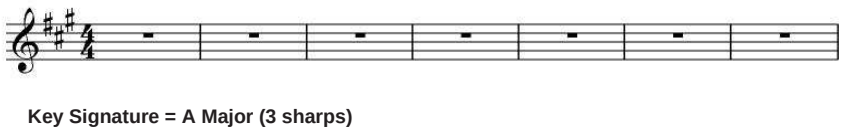

Figure 2.1: Key Signature

2.4 Scale

A *scale* is a graduated ascending (or descending) series of notes arranged in a specified order. A *scale degree* is a numeric position of a note within a scale ordered by increasing pitch. The simplest system is to name each degree after its numerical position in the scale, for example: the first (I), the second (II) etc.

2.4.1 Intervals

Notes in the scale are separated by whole and half step intervals of *tones* and *semitones*. Semitone is the interval between any note and the next note which may be higher or lower. Tone is the interval consisting of two semitones.

2.4.2 Equal temperament

Musically, the frequency of specific pitches is not as important as their relationships to other frequencies. The pitches of the notes in any given scale are usually related by a mathematical rule. Semitones are usually equally spaced out in a method known as *equal temperament*. Equal temperament is a scheme of musical tuning in which the octave is divided into a series of equal steps (equal frequency ratios). The best known example of such a system is *twelve-tone equal temperament* which is nowadays used in most Western music. Here, the pitch ratio between any two successive notes of the scale is exactly the twelfth root of two. So rare is the usage of other types of equal temperament, that the term “equal temperament ” is usually understood to refer to the twelve tone variety.

2.4.3 Chromatic Scale

The *chromatic* scale is a musical scale that contains all twelve pitches of the Western tempered scale. (C, C♯, D, D♯, E, F, F♯, G, G♯, A, A♯, B). In musical notation, sharp (♯) and flat (♭) mean higher and lower in pitch by a semitone respectively. The pitch ratio between any two successive notes of the scale is exactly the twelfth root of two. For convenience, we will use only the notation of sharps based on the *enharmonic equivalence* (identical in pitch) of sharps and flats. All of the other scales in traditional Western music are currently subsets of this scale.

2.4.4 Diatonic Scale

The *diatonic* scale is a fundamental building block of the Western musical tradition. It contains seven notes to the octave, made up of a root note and six other scale degrees. The list of names for the degrees of the scale are: *Tonic* (I), *Supertonic* (II), *Mediant* (III), *Subdominant* (IV), *Dominant* (V), *Submediant* (VI) and *Leading Tone* (VII). The Major and Minor scales are two

most commonly used diatonic scales and the term “diatonic” is generally used only in reference to these scales.

2.4.5 Major Scale

Tables 2.1 lists the pitch notes that are present in the 12 Major scales. Similar tables can be constructed for these scales with flats (♭) in them. The Major scale follows a pattern of: “T-T-S-T-T-T-S” on the twelve-tone equal temperament where T (implying Tone) and S (implying Semitone) corresponds to a jump of one and two pitch classes respectively. The elements of the Major Diatonic Scale corresponds to the Do, Rae, Me, Fa, So, La, Ti, Do (in order of scale degree) in *Solfege*, a pedagogical technique of assigning syllables to names of the musical scale.

Scale	Notes in Scale							
	I	II	III	IV	V	VI	VII	I
A	A	B	C♯	D	E	F♯	G♯	A
A♯	A♯	C	D	D♯	F	G	A	A♯
B	B	C♯	D♯	E	F♯	G♯	A♯	B
C	C	D	E	F	G	A	B	C
C♯	C♯	D♯	F	F♯	G♯	A♯	C	C♯
D	D	E	F♯	G	A	B	C♯	D
D♯	D♯	F	G	G♯	A♯	C	D	D♯
E	E	F♯	G♯	A	B	C♯	D♯	E
F	F	G	A	A♯	C	D	E	F
F♯	F♯	G♯	A♯	B	C♯	D♯	F	F♯
G	G	A	B	C	D	E	F♯	G
G♯	G♯	A♯	C	C♯	D♯	F	G	G♯

Table 2.1: Pitch notes in Major Scale

2.4.6 Minor Scales (Natural, Harmonic, Melodic)

Table 2.2 lists the pitch notes that are present in the 12 Minor Scales.

The Minor scales in Table 2.2 can be derived from the Major scales in Table 2.1. Every Major scale has a *Relative* Minor scale. The two scales are built from the exact same notes and the only difference between them is which note the scale starts with. The relative Minor scale

Scale	Notes in Scale							
	I	II	III	IV	V	VI	VII	I
Am	A	B	C	D	E	F	G	A
A♯m	A♯	C	C♯	D♯	F	F♯	G♯	A♯
Bm	B	C♯	D	E	F♯	G	A	B
Cm	C	D	D♯	F	G	G♯	A♯	C
C♯m	C♯	D♯	E	F♯	G♯	A	B	C♯
Dm	D	E	F	G	A	A♯	C	D
D♯m	D♯	F	F♯	G♯	A♯	B	C♯	D♯
Em	E	F♯	G	A	B	C	D	E
Fm	F	G	G♯	A♯	C	C♯	D♯	F
F♯m	F♯	G♯	A	B	C♯	D	E	F♯
Gm	G	A	A♯	C	D	D♯	F	G
G♯m	G♯	A♯	B	C♯	D♯	E	F♯	G♯

Table 2.2: Pitch notes in Minor Scale

starts from the sixth note of the Major scale. For example, the C Major scale is made up of the notes: “C-D-E-F-G-A-B-C” and its relative Minor scale, which is A Minor is made up of the notes “A-B-C-D-E-F-G-A”. A Minor is called the relative Minor of C Major, and C Major is the relative Major of A Minor. The relative Major/Minor combination for all the 12 pitch classes is illustrated in Table 2.3.

Major	C	C♯	D	D♯	E	F	F♯	G	G♯	A	A♯	B
Minor	A	A♯	B	C	C♯	D	D♯	E	F	F♯	G	G♯

Table 2.3: Relative Major and Minor Combinations

There is only one Major scale and three types of Minor scales for each of the 12 pitch classes. The Minor scale shown in Table 2.2 is the *Natural* Minor scale and what is simply referred to as the Minor scale. The *Harmonic* Minor scale is obtained by raising the VII note in the Natural Minor Scale by one semitone and the *Melodic* Minor scale is obtained by raising the VI note in addition to the VII note by one semitone. As an example, table 2.4 lists the notes that are present in all the 3 Minor Scales for C.

Scale	Notes in Scale							
	I	II	III	IV	V	VI	VII	I
Natural Minor	C	D	D♯	F	G	G♯	A♯	C
Harmonic Minor	C	D	D♯	F	G	G♯	B	C
Melodic Minor	C	D	D♯	F	G	A	B	C

Table 2.4: Notes in Minor scales of C

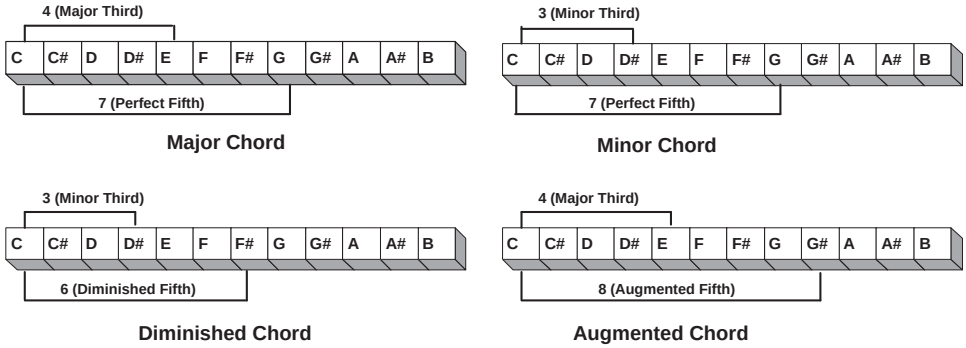

Figure 2.2: Types of Triads

2.5 Chords

Chord are a set of notes,usually with harmonic implication, played simultaneously. A *triad* is a chord consisting of 3 notes - a root, and two other members, usually a third and a fifth. The four types of triads shown in Figure 2.2 are:

- The *Major* chord contains four half steps between the root and the third (*a major third*), and seven half steps between the root and fifth (*a perfect fifth*). This is equivalent to the combination of the I, III and V note of the Major Scale.
- The *Minor* chord contains three half steps between the root and third (*a minor third*), and the same perfect fifth between the root and fifth. This is equivalent to the combination of the I, III and V note of the Minor Scale.
- The *Diminished* chord contains three half steps between the root and third (a minor third), and six half steps between the root and fifth (*a diminished fifth*)
- The *Augmented* chord consists of four half steps between the root and the third (major third) and eight half steps between the root and the fifth (*an augmented fifth*)

There are only 2 kinds of keys possible : Major and Minor; and the chord patterns built on the 3 Minor scales (Natural, Harmonic and Melodic) are all classified as being simply in the Minor key. Thus we have 12 Major and 12 Minor keys (henceforth referred to as 24 Major/Minor

keys).

Table 2.5 shows the chord patterns in Major and Minor keys. Roman numerals are used to denote the scale degree. Upper case roman numerals indicate Major chords, lower case roman numerals refer to Minor chords, ● indicates a Diminished chord and the + sign indicates an Augmented chord. These chords are obtained by applying the interval patterns of Major, Minor, Diminished and Augmented chords discussed earlier in this section.

Key	**Chords**							
Major	I	ii	iii	IV	V	vi	vii●	I
Natural Minor	i	ii●	III	iv	v	VI	VII	i
Harmonic Minor	i	ii●	III+	iv	V	VI	♯vii●	i
Melodic Minor	i	ii	III+	IV	V	♯vi●	♯vii●	i

Table 2.5: Chords in Major and Minor Keys

As an example, Table 2.6 shows the chords in the Major and Minor key of C. It is observed that the chord built on the third note of the Natural Minor scale is D♯ Major. This is obtained by extracting the 1-3-5 elements on the D♯ Natural Minor scale - D♯, G and A♯. This corresponds with the interval pattern for the D♯ Major chord.

Key	**Chords**							
	I	II	III	IV	V	VI	VII	I
Major	C maj	D min	E min	F maj	G maj	A min	B dim	C maj
N. Minor	C min	D dim	D♯ maj	F min	G min	G♯ maj	A♯ maj	C min
H. Minor	C min	D dim	D♯ aug	F min	G maj	G♯ maj	B dim	C min
M. Minor	C min	D min	D♯ aug	F maj	G maj	A dim	B dim	C min

Table 2.6: Chords in Major and Minor Key for C

Chapter 3

System Description

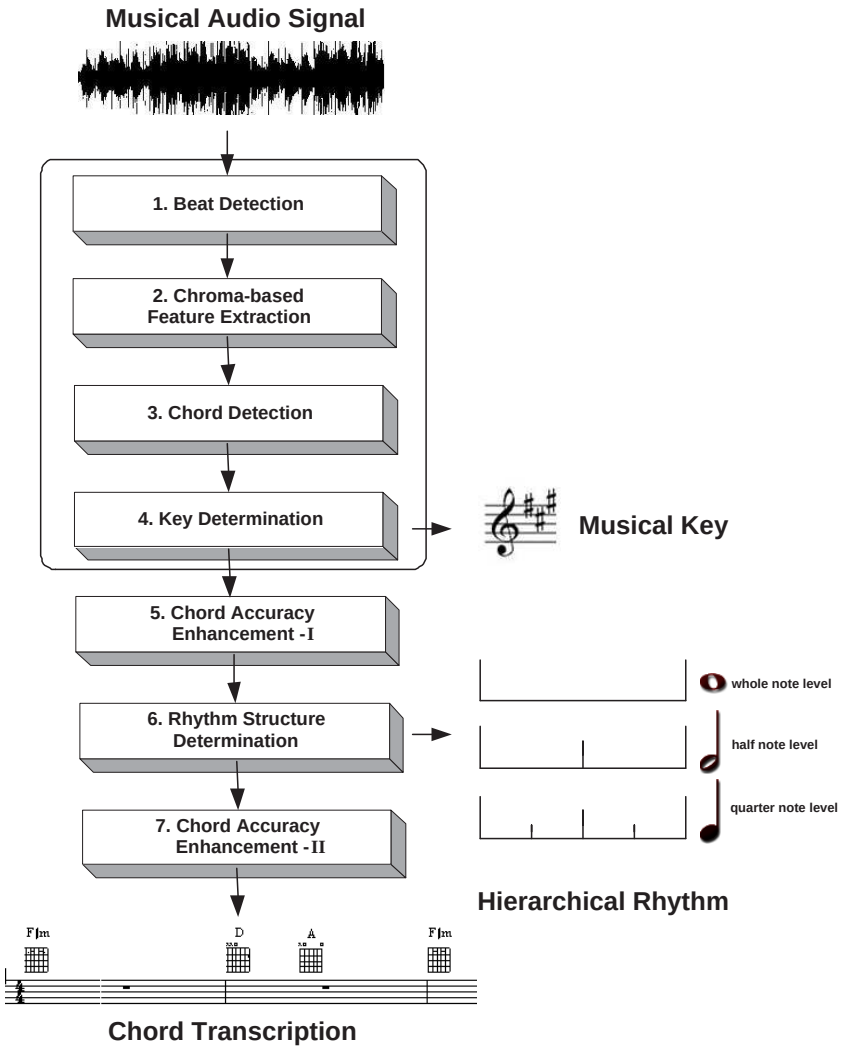

Figure 3.1: System Components

The block diagram of the proposed framework is shown in Figure 3.1. We draw on the prior idea of Goto and Muraoka in [15, 17] to incorporate higher level music knowledge of the relation between rhythm and chord change patterns. Our technique is based on a combination of bottom-up and top-down approaches, combining the strength of low-level features and high-level musical knowledge.

Our system seeks to perform a music-theoretical analysis of an acoustic musical signal and output the musical key, harmonic description in the form of the 12 Major and 12 Minor triad chords (henceforth referred to as the 24 Major/Minor triads) and the hierarchical rhythm structure at the quarter note, half note and whole note (measure) levels.

The first step in the process is the detection of the musical key. A well known algorithm used to identify the key of the music is called the *Krumhansl-Schmuckler key-finding algorithm* which was developed by Carol Krumhansl and Mark Schmuckler [22]. The basic principle of the algorithm is to compare a prototypical Major (or Minor) scale-degree profile (individual notes within a scale ordered by increasing pitch) with the input music. In other words, the distribution of pitch-classes in a piece is compared with an ideal distribution for each key. Several enhancements to the basic algorithm have been suggested in [20, 48, 49].

For input, the algorithm above uses an input vector which is weighted by duration of the pitch classes in the piece. It requires a list of notes with ontimes and offtimes. However, in the audio domain, overlap of harmonic components of individual notes in real-world musical recordings would make it a difficult task to determine the actual notes or their duration. A large number of notes are detected in the frequency analysis. Hence the algorithm cannot be directly applied.

Thus we have approached this problem at a higher level by clustering individual notes detected and have tried to obtain the harmonic description of the music in the form of the 24 Major/Minor triads. Then based on a rule-based analysis of these chords against the chords present in the Major and Minor keys, we extract the key of the song.

However, the chord recognition accuracy of the system, though sufficient to determine the key, is not sufficient to provide usable chord transcriptions or determine the hierarchical rhythm structure across the entire length of the music. We have thus enhanced the four-step key determination system with three postprocessing stages that allow us to perform these two tasks with greater accuracy, as shown in the Figure 3.1. In the next section the seven individual components of this framework are discussed.

Chapter 4

System Components

4.1 Beat Detection

According to Copland in [9], *rhythm* is one of the four essential elements of music. Music unfolds through time in a manner that follows rhythm structure. Measures of music divide a piece into time-counted segments and time patterns in music are referred to in terms of meter. The beat forms the basic unit of musical time and in a meter of 4/4 (known as common time or quadruple time) there are four beats to a measure. Rhythm can be perceived as a combination of strong and weak beats. A strong beat usually corresponds to the first and third quarter note in a measure and the weak beat corresponds to the second and fourth quarter note in a measure [16]. If the strong beat constantly alternates with the weak beat, the inter-beat-interval (the temporal difference between two successive beats), would correspond to the temporal length of a quarter note. For our purpose, the strong and weak beat as defined above, corresponds to the alternating sequence of equally spaced phenomenal impulses which define the tempo for the music [41]. We assume the meter to be 4/4, this being the most frequent meter of popular songs and the tempo of the input song is assumed to be constrained between 40-185 M.M. (M$\ddot{a}$lzels Metronome: the number of quarter notes per minute) and almost constant.

Our system aims to extract rhythm information in real world musical audio signals in the form of a hierarchical beat-structure representation comprising the quarter note, half note, and whole note or measure levels. As a first step towards this end, the musical signal is framed into beat-length segments to extract metadata in the form of quarter note detection of the music. The basis for this technique of audio framing is to assist in the detection of chord structures in the music based on the following knowledge of chords [17]:

- $Premise_1$: Chords are more likely to change on beat times than on other positions.
- $Premise_2$: Chords are more likely to change on half note times than on other positions of beat times.
- $Premise_3$: Chords are more likely to change at the beginning of the measures than at other positions of half note times.

Our beat detection process first detects the onsets present in the music using sub-band processing [52]. This technique of onset detection is based on the sub-band intensity to detect the perceptually salient percussive events in the music signal. We draw on the prior ideas of beat tracking discussed in [11, 41] to determine the beat structure of the music as follows:

1. Compute all possible values of inter-onset intervals (IOIs). An IOI is defined as the time interval between any pair of onsets, not necessarily successive.
2. Compute clusters of IOIs and create a ranked set of hypothetical inter-beat-intervals (IBIs) based on the size of the corresponding clusters and by identifying integer relationships with other clusters. The latter is to recognize harmonic relationships between the beat (quarter note level) and simple integer multiples of the beat (half note and whole note levels). An error margin of $\pm$ 25 ms has been set in the IBI to account for slight variations in the tempo.

3. The hightest ranked value is returned as the IBI from which we obtain the tempo, expressed as an inverse value of the IBI.

4. Track patterns of onsets in clusters at the IBI and interpolate beat information in sections where onsets corresponding to the beat might not be detected.

Let $\mathcal{T}$ = $\{t_1, t_2,...t_n\}$ % be the set of all detected transients (onsets)
Let $\mathcal{Q}$ = $\{q : 325 \leq q \leq 1500\}$ % be the set of all possible quarter-note intervals
Let $\mathcal{IOI}$ = $\{ioi_q : \forall\ q \in \mathcal{Q}\}$ % maintain the count of all values of inter-onset-intervals

for i = 1... (n-1)
Begin
  for j = (i+1)...n
  Begin
    $ioi_{(t_j - t_i)}$ = $ioi_{(t_j - t_i)}$ + 1
  End
End

Let $\mathcal{D}$ = {d: d $\in \mathcal{Q}$, $ioi_d \in$ 4 largest elements in $\mathcal{IOI}$ }
% be the set containing the 4 largest cluster size values

$\forall\ q \in \mathcal{Q}$ % to identify harmonic relationships between hypothetical
% beat value (quarter note) and simple integer multiples
Begin % (half note and whole note)

  If ($ioi_q \in \mathcal{D}$) and ($\exists\ ioi_{2q}$, $ioi_{4q} \in \mathcal{D}$ such that $ioi_q \approx$ (2 *ioi_{2q}) $\approx$ (4*ioi_{4q}))
  Begin
    $Quarter\_Note$ = q
  End
End

$Tempo$ = 60,000 / $Quarter\_Note$
$Beat\_Sequence$ = $t_1 \rightarrow t_2 \rightarrow t_3$... such that t_k - t_{k-1} = $Quarter\_Note \pm 25\ ms$

Table 4.1: Beat Detection Algorithm

The algorithm has been highlighted in Table 4.1. The results of our tempo detection and beat structure detection is shown in Figure 4.1 and 4.2 respectively for the song "*Back to you*" by *Bryan Adams*.

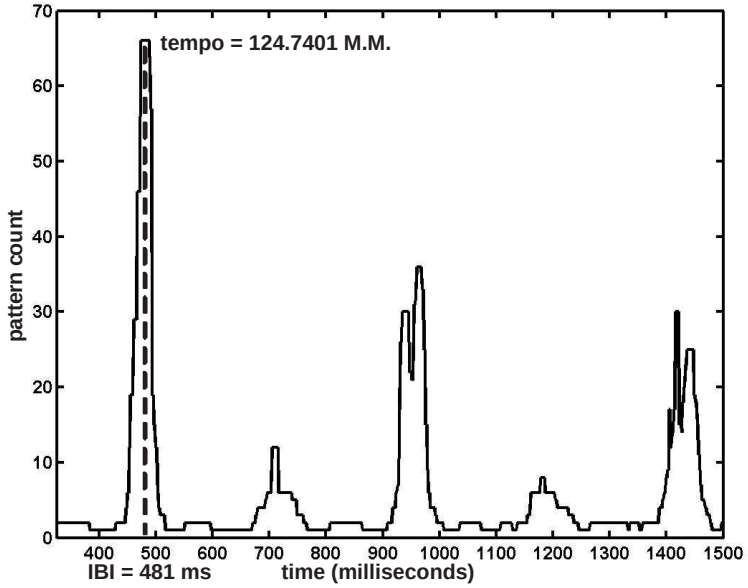

Figure 4.1: Tempo Detection

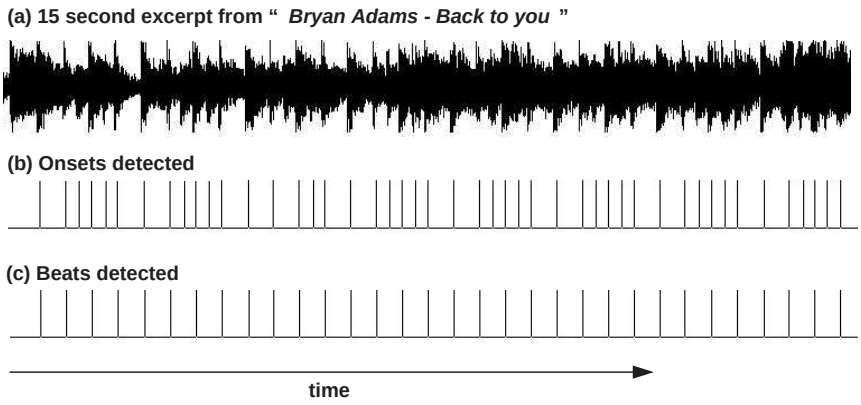

Figure 4.2: Beat Detection

4.2 Chroma Based Feature Extraction

As highlighted in [4], there are two distinct attributes of pitch perception, *Tone Height* and *Chroma* [45]. Tone Height describes the general increase in the pitch of a sound as its frequency increases. Chroma, on the other hand, is cyclic in nature with octave periodicity. Chroma is closely related to the theoretical concept of pitch class. Under this formulation two tones separated by an integral number of octaves share the same value of Chroma. Later, it has been suggested that one could decompose frequency into similar attributes [32].

The feature which we are using is a reduced spectral representation of each beat-spaced segment of the audio based on a Chroma transformation of the spectrum. This feature class represents the spectrum in terms of pitch-class, and forms the basis for the *Chromagram* [51].

The input signal is transformed into the frequency domain. For each quarter-note spaced segment of audio, this is then restructured into a Chroma spectrum by summing and normalizing the pitch chroma intensities over 5 octaves using the frequencies of pitch notes in the tempered scale [3] as shown in Table 4.2. This mapping procedure provides us with a highly reduced representation of the frame, consisting of a single 12-element feature vector corresponding to the 12 pitch classes.

Octave	**C-2 to B-2**	**C-3 to B-3**	**C-4 to B-4**	**C-5 to B-5**	**C-6 to B-6**
C	65.406	130.813	261.626	523.251	1046.502
C♯	69.296	138.591	277.183	554.365	1108.730
D	73.416	146.832	293.665	587.330	1174.659
D♯	77.782	155.563	311.127	622.254	1244.508
E	82.407	164.814	329.628	659.255	1318.510
F	87.307	174.614	349.228	698.456	1396.913
F♯	92.499	184.997	369.994	739.989	1479.978
G	97.999	195.998	391.995	783.991	1567.982
G♯	103.826	207.652	415.305	830.609	1661.219
A	110.000	220.000	440.000	880.000	1760.000
A♯	116.541	233.082	466.164	932.328	1864.655
B	123.471	246.942	493.883	987.767	1975.533

Table 4.2: Musical Note Frequencies

We have found it useful to employ the musical relevance of Chroma in the development of features for our purpose since various 3-element pitch class combinations in the Chroma vector can be used to detect the presence of Major and Minor chords in an audio frame.

4.3 Chord Detection

In this current work, we have considered only the Major and Minor triads. This is because they are the most commonly used chords in western music and constitute the majority of the chords for any key as can be seen from tables 2.5 and 2.6

For our analysis we consider only the elements with the four highest values in the Chroma vector and assign weights to them accordingly. Four elements are sufficient to distinguish between a Major and Minor chord. This is because they share the same Tonic (I) and Dominant note (V) and differ only in the position of the Mediant note (III). For a Minor chord, it is one semitone lower than the one for the Major chord. For example, the C Major chord is comprised of C, E and G notes and the C Minor chord is comprised of C, D♯ and G notes. This is illustrated with an example in Figure 4.3 and the algorithm is highlighted in Table 4.3

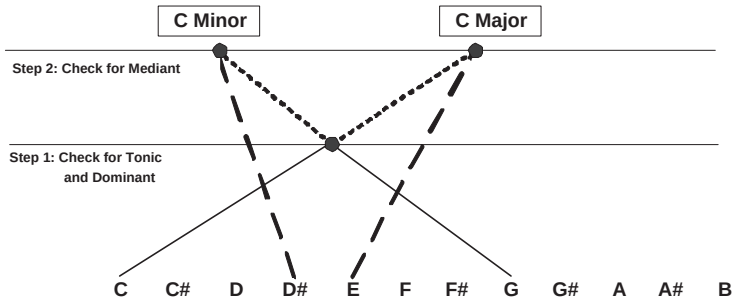

Figure 4.3: Chord Detection Example

The chords detected across all the beat-spaced frames are then used to create a histogram, a 24 element vector whose elements correspond to the 24 Major/Minor triads. This will be used for key determination in the next stage.

Let $\mathcal{C}$ = {C, C♯, D,..... B} be the 12 elements of the chromatic scale

Let $\mathcal{S}$ = {S_C, $S_{C\sharp}$, S_D,..... S_B } be the signal strengths of the individual pitch notes (Chroma Vector)

Let $\mathcal{D}$ = {d: d $\in \mathcal{C}$, $S_d \in$ 4 largest elements in $\mathcal{S}$ }

$\forall$ c $\in \mathcal{C}$, set Tonic = c % perform loop 12 times

Begin

Dominant = c + 7 semitones % perfect fifth interval

Mediant_Minor = c + 3 semitones % minor third interval

Mediant_Major = c + 4 semitones % major third interval

If (Tonic & Dominant) $\in \mathcal{D}$

Begin

Case : (Mediant_Major & Mediant_Minor) $\in \mathcal{D}$

Chord = c(Major) if $S_{Mediant\_Major} \geq S_{Mediant\_Minor}$

Chord = c(Minor) if $S_{Mediant\_Major} < S_{Mediant\_Minor}$

Case : Mediant_Major $\in \mathcal{D}$

Chord = c(Major)

Case : Mediant_Minor $\in \mathcal{D}$

Chord = c(Minor)

End

End

Table 4.3: Chord Detection Algorithm

It is to be noted that complexities in polyphonic audio analysis often result in chord recognition errors. Thus we are unable to obtain usable chord transcriptions at this stage.

4.4 Key Determination

As highlighted in section 1.1, the key defines the diatonic scale which a piece of music uses. The diatonic scale is a seven note scale and is most familiar as the Major/Minor scale in music. *Tonic/tonality* is sometimes used interchangeably with key. Tonality is an important structural property of music, and has been described by music theorists and psychologists as a hierarchical ordering of the pitches of the chromatic scale such that these notes are perceived in relation to one central and stable pitch, the tonic [46]. This hierarchical structure is manifest in listeners' perceptions of the stability of pitches in tonal contexts.

The Krumhansl-Schmuckler Key-Finding Algorithm and its variations described in section 3 cannot be directly applied to polyphonic audio as it requires a list of notes with ontimes and offtimes which cannot be directly extracted from polyphonic audio. Hence we introduce the concept of musical key determination at this stage that serves two purposes:

1. Identify the diatonic scale, and hence the individual notes that a piece of music uses: This process will use the chords detected thus far (correct and wrong) to categorize a given music signal into one of the 24 Major/Minor keys.

2. Perform error correction on the detected chords: Complexities in polyphonic audio analysis often results in chord recognition errors. Knowledge of the key will allow us to identify the erroneous chords among the chords detected via music-theoretic analysis. We can then define a criterion to eliminate them as will be discussed in the next section.

In this process, the 24 element vector of Major and Minor chords created in the previous step is pattern matched using weighted Cosine Similarity against 24 element reference vectors created for each of the 24 Major/Minor keys. The pattern that returns the highest rank is selected as the one being the key of the song. We assume the key to be constant throughout the length of

the song. The algorithm has been highlighted in Table 4.4

Let w_{ij}, i , j = 1...24 be the 24 element reference vectors for the 24 Major / Minor keys Let v_i, i = 1..24 be the 24 element input vector Select key = j $\quad \mid \cos\Theta = \frac{\sum_{i=1}^{24} v_i w_{ij}}{\sqrt{\sum_{i=1}^{24}(v_i)^2}\sqrt{\sum_{i=1}^{24}(w_{ij})^2}}$ is max $\forall$ j = 1...24

Table 4.4: Key Determination Algorithm

An important point to be noted here is that the similarity analysis has been biased by assigning relatively higher weights to the *primary chords* in each key. The system of primary chords was formulated by the French composer *Jean Phillipe Rameu* in the 18th century in his book *Treatise on Harmony*. The primary chords are the three most important chords in a key and every note of the scale is part of at least one of the primary chords. The first is the triad built on the root or tonic note, and it is called the *root* or the *tonic chord*. The next is the chord built on the fifth note, called the *dominant chord*. The third chord is built on the fourth note, and is called the *subdominant chord*. In the key of C Major these chords are C Major, G Major and F Major respectively.

The primary chords for each key can be determined using the chord patterns in Table 2.5 and 2.6. A simpler way of approaching this would be to use the *circle of fifths*. The circle of fifths, shown in Figure 4.4 is a visualization of relations between keys. In the circle of fifth the three primary chords are always next to each other: The tonic or root in the center, the subdominant to the left (counterclockwise) and the dominant to the right (clockwise). For example: In the key of C Major, C is on the top, the subdominant F is to the left, and the dominant G is to the right. The notes on the outside of the circle represent the Major keys and those on the inside are all the relative Minor keys.

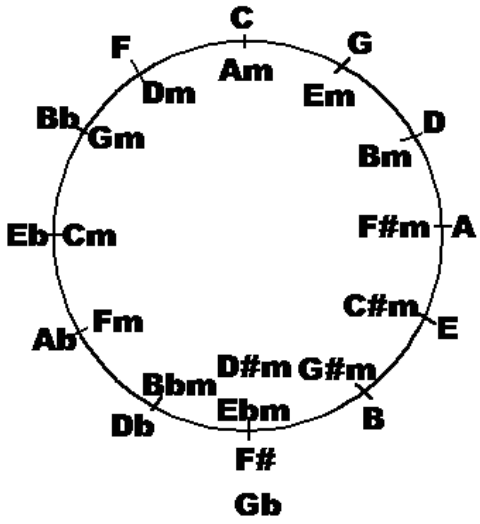

Figure 4.4: Circle of Fifths

4.5 Chord Accuracy Enhancement - I

In this step we aim to increase the accuracy of chord detection. For each audio frame:

- [**Check 1**] **Eliminate erroneous chords not in the key of the song:**

  Perform a rule-based analysis of the detected chord to see if it exists in the key of the song. If it does not:

  a. Check for the presence of the Major chord of the same pitch class if the detected chord is a Minor and vice-versa. If this is present in the key, replace the erroneous chord with this chord. This is because the Major and Minor chord of a pitch class differ only in the position of the Mediant note. The chord detection approach often suffers from recognition errors that result from overlaps of harmonic components of individual notes in the spectrum that is quite difficult to avoid. Hence there is a possibly of error in the distinction between the Major and Minor chords for a given pitch class.

  b. If the check fails, eliminate the chord.

- [**Check 2**] **Perform temporal corrections of detected or missing chords:**

  If the chords detected in the adjacent frames are the same but different from the current

frame, then the chord in the current frame is likely to be incorrect. In these cases, we coerce the current frame's chord to match the one in the adjacent frames.

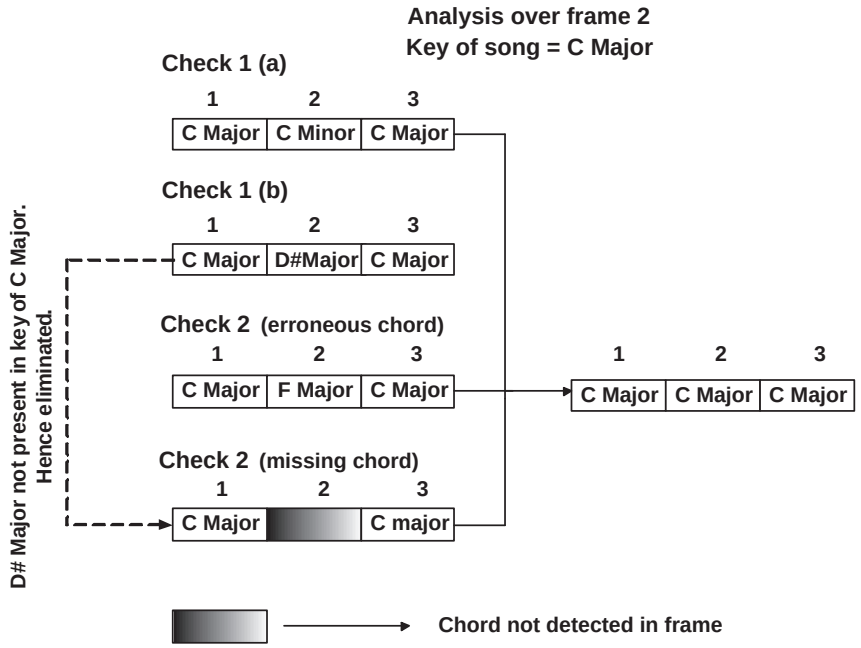

Figure 4.5: Chord Accuracy Enhancement - I

We present an illustrative example of the above checks over three consecutive quarter note spaced frames of audio in Figure 4.5

4.6 Rhythm Structure Determination

Check for start of measures based on the premise that chords are more likely to change at the beginning of a measure than at other positions of beat times [15]. Since there are 4 quarter notes to a measure, check for patterns of 4 consecutive frames with the same chord to demarcate all the possible measure boundaries. However not all of these boundaries may be correct. We will illustrate this with an example in which a chord sustains over 2 measures of the music.

From Figure 4.6(c), it can be seen that there are 4 possible measure boundaries being de-

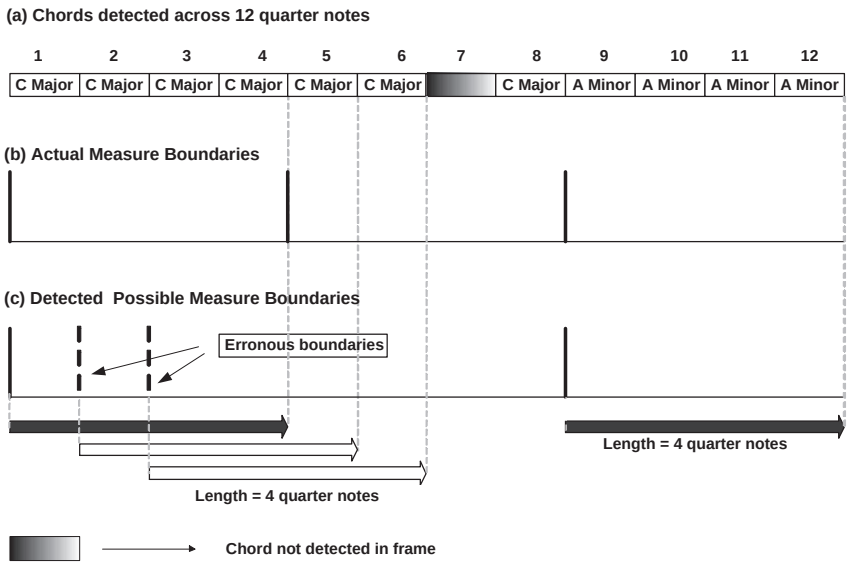

Figure 4.6: Error in Measure Boundary Detection

tected across the 12 quarter note spaced frames of audio. Our aim is to eliminate the 2 erroneous ones (dotted line in Figure 4.6(c)) and interpolate an additional measure line at the start of the fifth frame to give us the required result as seen in Figure 4.6(b). The correct measure boundaries along the entire length of the song are thus determined as follows:

1. Along the increasing order on the time axis, obtain all possible patterns of boundaries originating from every boundary location that are have integer relationships in multiples of 4. Select the pattern with the highest count as the one corresponding to the pattern of actual measure boundaries.

2. Track the boundary locations in the detected pattern and interpolate missing boundary positions across the rest of the song.

The result of our hierarchical rhythm detection is shown in Figure 4.7. The symbolic representation of the hierarchical rhythm structure in Figure 4.7(d) can be interpreted as shown in Figure 3.1.

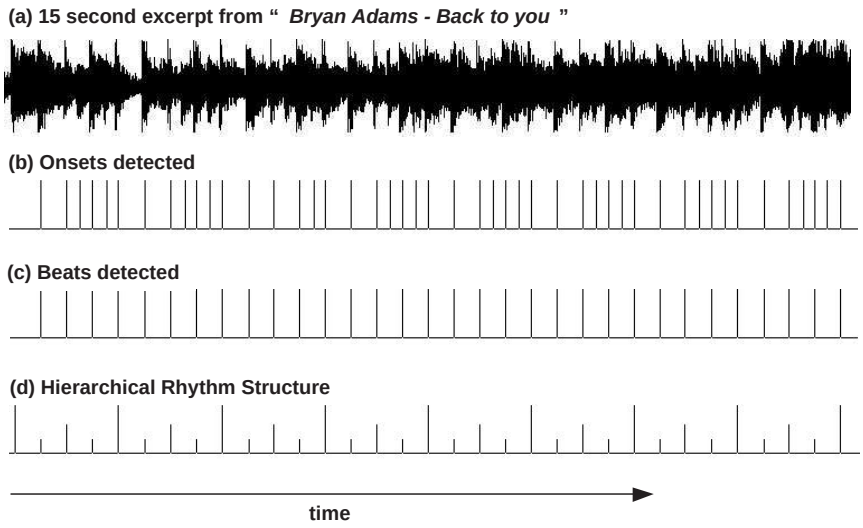

Figure 4.7: Hierarchical Rhythm Structure

4.7 Chord Accuracy Enhancement - II

Now that the measure boundaries have been extracted, we can increase the chord accuracy in each measure of audio as follows:

- [**Check 3**] **Intra-measure Chord Check:**

  From $Premise_3$, we know that chords are more likely to change at the beginning of the measures than at other positions of half note times. Hence:

  a. If three of the chords are the same, then the 4th chord is likely to be the same as the others.

  b. If there is a chord common to both halves of the measure, then all the chords in the measure are likely to be the same as this chord.

It is observed that all possible cases of chords under $[Check3]$ (a) are already handled by $[Check1, 2]$ above. Hence we only implement $[Check3]$(b) and this is illustrated in Figure 4.8 with an example. This check is required because, in the case of a Minor key, we can have both the Major and Minor chord of the same pitch class present in the song. A classic example of this can be seen in "*Hotel California*" by the *Eagles* . This song is in the key of B Minor and the

chords in the verse include an E Major and an E Minor chord which shows a musical shift from the Melodic Minor to the Natural Minor. Here if an E minor is detected in a measure containing the E major chord, $[Check1]$ would not detect any error on account of both the E Major and E Minor chord being present in the key of B Minor.

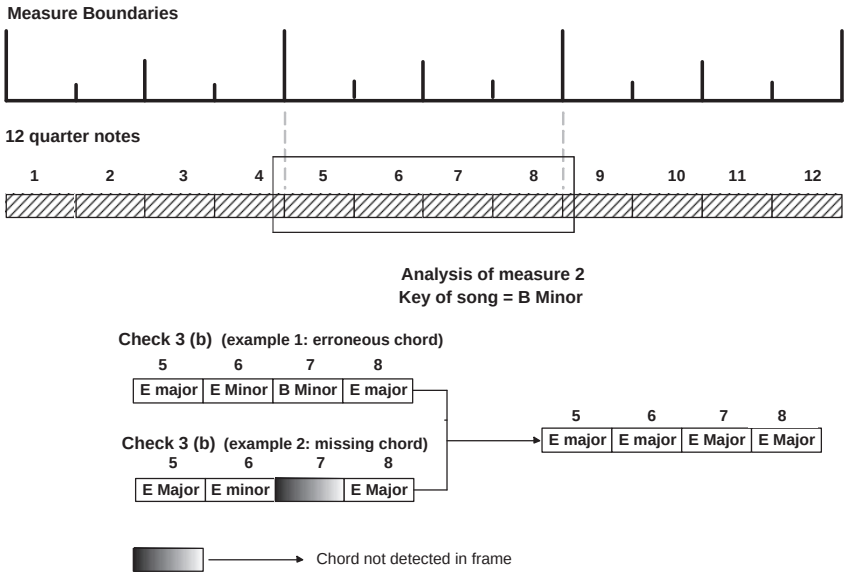

Figure 4.8: Chord Accuracy Enhancement - II

Chapter 5

Experiments

5.1 Results

The results of our experiments, performed on 30 popular English songs spanning 5 decades of music are tabulated in Table 5.1. The songs have been carefully selected for their variety in artist and time spans.

It can be observed that the average chord detection accuracy across the length of the entire music performed by the Chord Detection step (module 3 in our framework) is relatively low at 48.13%. The rest of the chords are either not detected or detected in error. The latter is reflected primarily by the difference between A & B in Table 5.1 as B performs the correction or elimination of erroneous chords not in the key of the song.

This accuracy is however sufficient to determine the key accurately for 28 out of 30 songs in the Key Detection step (module 4 in our framework) which reflects an accuracy of over 93% for key detection. This has verified against the information in the commercially available sheet music for the songs, a good source of which can be found at [26, 43].

No.	Song Title	A (%)	Original Key	Detected Key	B (%)	Measure Detection	C (%)
1	(1965) Righteous Brothers - Unchained melody	57.68	C maj	C maj	70.92	Yes	85.11
2	(1977) Bee Gees - Stayin' Alive	39.67	F min	F min	54.91	Yes	71.40
3	(1977) Eric Clapton - Wonderful tonight	27.70	G maj	G maj	40.82	Yes	60.64
4	(1977) Fleetwood Mac - You make lovin' fun	44.37	A♯ maj	A♯ maj	60.69	Yes	79.31
5	(1979) Eagles - I can't tell you why	52.41	D maj	D maj	68.74	Yes	88.97
6	(1984) Foreigner - I want to know what love is	55.03	D♯ min	D♯ min	73.42	No	58.12
7	(1986) Bruce Hornsby - The way it is	59.74	G maj	G maj	70.32	Yes	88.50
8	(1989) Chris Rea - Road to hell	61.51	A min	A min	76.64	Yes	89.24
9	(1991) R.E.M. - Losing my religion	56.31	A min	A min	70.75	Yes	85.74
10	(1991) U2 - One	56.63	C maj	C maj	64.82	Yes	76.63
11	(1992) Michael Jackson - Heal the world	30.44	A maj	A maj	51.76	Yes	68.62
12	(1993) MLTR - Someday	56.68	D maj	D maj	69.71	Yes	87.30
13	(1995) Coolio - Gangsta's paradise	31.75	C min	C min	47.94	Yes	70.79
14	(1996) Backstreet Boys - As long as you love me	48.45	C maj	C maj	61.97	Yes	82.82
15	(1996) Joan Osborne - One of us	46.90	A maj	A maj	59.30	Yes	80.05
16	(1997) Bryan Adams - Back to you	68.92	C maj	C maj	75.69	Yes	95.80
17	(1997) Green Day - Time of your life	54.55	G maj	G maj	64.58	Yes	87.77
18	(1997) Hanson - Mmmbop	39.56	A maj	A maj	63.39	Yes	81.08
19	(1997) Savage Garden - Truly, madly, deeply	49.06	C maj	C maj	63.88	Yes	80.86
20	(1997) Spice Girls - Viva forever	64.50	D♯ min	F♯ maj	74.25	Yes	91.42
21	(1997) Tina Arena - Burn	35.42	G maj	G maj	56.13	Yes	77.38
22	(1998) Jennifer Paige - Crush	40.37	C♯ min	C♯ min	55.41	Yes	76.78
23	(1998) Natalie Imbruglia - Torn	53.00	F maj	F maj	67.89	Yes	87.73
24	(1999) Santana - Smooth	54.53	A min	A min	69.63	No	49.91
25	(2000) Corrs - Breathless	36.77	B maj	B maj	63.47	Yes	77.28
26	(2000) Craig David - Walking away	68.99	A min	C maj	75.26	Yes	93.03
27	(2000) Nelly Furtado - Turn off the light	36.36	D maj	D maj	48.48	Yes	70.52
28	(2000) Westlife - Seasons in the sun	34.19	F♯ maj	F♯ maj	58.69	Yes	76.35
29	(2001) Shakira - Whenever, wherever	49.86	C♯ min	C♯ min	62.82	Yes	78.39
30	(2001) Train - Drops of Jupiter	32.54	C maj	C maj	53.73	Yes	69.85
	Average accuracy at each stage	**48.13**	**28/30 songs**		**63.20**	**28/30 songs**	**78.91**
	(A ⟶ Chord Detection) (B ⟶ Chord Accuracy Enhancement-I) (C ⟶ Chord Accuracy Enhancement-II)						

Table 5.1: Experimental Results

The average chord detection accuracy of the system improves on an average by 15.07% on applying Chord Accuracy Enhancement - I (module 5 in our framework). Errors in key determination do not have any effect on this step as will be discussed in Section 5.2. The new accuracy of 63.20% has been found to be sufficient to determine the hierarchical rhythm structure (module 6 in our framework) across the music for 28 out of the 30 songs, thus again reflecting an accuracy of over 93% for rhythm tracking.

Finally the application of Chord Accuracy Enhancement - II (module 7 in our framework)

makes a substantial performance improvement of 15.71% leading to a final chord detection accuracy of 78.91%. This could have been higher, were it not for the performance drop for the 2 songs (6 and 24 in table 5.1) on account of error in measure boundary detection. This exemplifies the importance of accurate measure detection in order to perform intra-measure chord checks based on the previously discussed musical knowledge of chords.

5.2 Key Determination Observation

It can be observed that for 2 of the songs (song numbers, 20 and 26 in Table 5.1), the key has been determined incorrectly. The explanation for this can be based on the theory of the Relative Major/Minor combination of keys as explained earlier in this paper.

Our technique assumes that the key of the song is constant throughout the length of the song. However, many songs often use both Major and Minor keys, perhaps choosing a Minor key for the verse and a Major key for the chorus, or vice versa. This has a nice effect, as it helps break up the monotony that sometimes results when a song lingers in one key. Often, when switching to a Major key from a Minor key, the songwriters will choose to go to the Relative Major from the Minor key the song is in and vice-versa. Sometimes the chords present in the song are present in both the Major and its relative Minor.

For example, the 4 main chords used in the song "*Viva Forever*" by the *Spice Girls* are - D♯ Minor, A♯ Minor, B Major and F♯ Major. These chords are present in the key of F♯ Major and D♯ Minor. Hence it is difficult for the system to determine if the song is in the Major key or the Relative minor. This can be taken as a probable explanation for both the songs with erroneous key results where the relative Major has been detected instead of the actual Minor key. A similar observation can be made for the song "*Walking Away*" by *Craig David* where the main chords used are - A Minor, D Minor, F Major, G Major and C Major. These chords are present in both,

the key of C Major as well as in it's relative Minor, A Minor.

The usage of weighted Cosine similarity technique causes the shorter Major key patterns to be preferred over the longer Minor key patterns. This is because of the normalization that is performed while applying the Cosine similarity. For the Minor keys, normalization is applied taking into account the count of chords that can be constructed across all the three types of Minor scales. However, from an informal evaluation of popular music we observe that popular music in Minor keys usually shift across only two out of the three scales, primarily the Natural and Harmonic Minor. So in such cases, the normalization technique applied would cause the system to get slightly biased towards the relative Major key where this problem is not present as there is only one Major scale.

Errors in key recognition, does not affect the Chord Accuracy Enhancement - I because we also consider chords present in the relative Major/Minor key in addition to the chords in the detected key. A theoretical explanation on how to perform key identification in such cases of ambiguity (as seen above) based on an analysis of sheet music can be found in [12].

5.3 Chord Detection Observation

The variation in the chord detection accuracy of the system can be explained as follows:

1. **Usage of other chords:**

   In this approach we have considered only the Major and Minor triads. However, in addition to these, there are other chord possibilities in popular music as highlighted in [23]. Of particular interest to us here is the Dominant 7^{th} category of chords as discussed by Jazz legend, *Joe Pass*, in [31]. Under this formulation, the Augmented and Diminished chords are included in the Dominant 7^{th} category. The usage of chords from the Dominant 7^{th} category varies in commercial music and may be the reason for variation in chord

detection.

2. **Presence of extended chords:**

   It is quite common to see *extended* chords in music. Extended chords are chords obtained by adding diatonic intervals to the basic Major and Minor triads to add "color" to the basic chord. For example:

   C Major7 Chord (C E G B) = C Major triad + VII degree of C Major scale
   C Minor7 Chord (C D# G A#) = C Minor triad + VII degree of C Major scale

   Polyphony, with its multidimensional sequences of overlapping tones and overlapping harmonic components of individual notes in the spectrum might cause the elements in the Chroma vector to be weighted wrongly. So a C Major 7 chord (C E G B) in the music might wrongly get detected as an E Minor chord (E G B) if the latter 3 notes are assigned a relatively higher weight in the Chroma vector.

3. **Key Change:**

   In some songs, there is a key change toward the end of a song to make the final repeated part(s) (chorus/refrain) slightly different from the previous parts. This is effected by transposing the song to higher semitones, usually up a half step. This has also been highlighted by Goto in [14]. Since our system does not currently handle key changes, the chords detected in this section will not be recognized. This is illustrated with an example in Figure 5.1

Another point to be noted here is of chord substitution/ simplification of extended chords in the evaluation of our system. For simplicity, extended chords can be substituted by the respective Major/Minor triad. As long as the notes in the extended chord are present in the scale, and the basic triad is there, the simplification can be done. For example, The C Major7 can be simplified to the C Major triad. This substitution has been performed on the extended chords annotated in

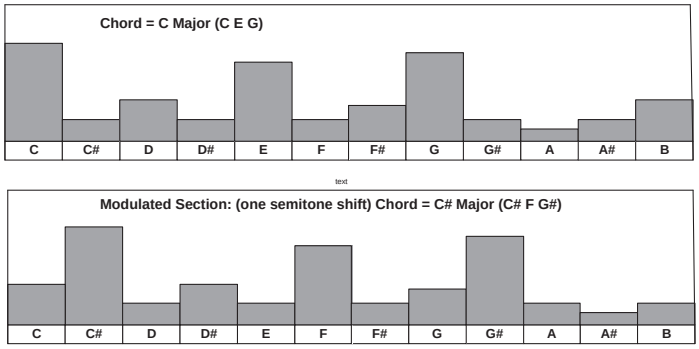

Figure 5.1: Key Modulation

the sheet music in the evaluation of our system.

5.4 Rhythm Tracking Observation

We have observed two reasons for error in measure detection:

1. Two patterns of measure boundaries detected by our rhythm detector have the same length. Hence, our system is unable to make a decision on which pattern to select.

2. An incorrect pattern of measure boundaries has the highest count.

We conjecture this to be on account of errors in the chord detection as discussed above. Chords present in the music and not handled by our system could be wrongly classified into one of the 24 Major/Minor triads on account of complexities in polyphonic audio analysis. This can result in incorrect clusters of 4 chords being captured by the rhythm detection process.

Further, beat detection is a non-trivial task and the difficulties of tracking the beats in acoustic signals are discussed in [16]. Any error in beat detection can cause a shift in the rhythm structure determined by the system.

Chapter 6

Conclusion

We have presented a technique to determine the key, chords and hierarchical rhythm structure from acoustic musical signals. To our knowledge, this is the first attempt to use a rule-based approach that combines low-level features with high level music knowledge of rhythm and harmonic structure to determine all three of these expressive dimensions of music.

We have demonstrated the applicability of this framework in various other aspects of content analysis like singing voice detection and the automatic alignment of textual lyrics and musical audio. (Appendix A : Publications)

The human auditory system is capable of extracting rich and meaningful data from complex audio signals [44] and existing computational auditory analysis systems fall clearly behind humans in performance. Towards this end, we believe that the model proposed here, provides a promising platform for the future development of more sophisticated auditory models based on a better understanding of music. Our current and future research that builds on this work is highlighted below:

- **Key Detection :** Our technique assumes that the key of the song is constant throughout the length of the song. However, on account of the properties of the relative Major/Minor

key combination, we have made the chord and rhythm detection process (that uses the key of the song as input), quite robust against changes across this key combination. However, the same cannot be said about other kinds of key changes in the music. This is because such key changes are quite difficult to track since there are no fixed rules and depend more on the songwriter's creativity. For example, the song "*Let It Grow*" by *Eric Clapton* switches from a B Minor key in the verse to an E Major key in the chorus. We believe that an analysis of the song structure (verse, chorus, bridge etc.) could probably serve as an input to track these kind of key changes. This problem is currently being analyzed and will be tackled in the future.

- **Chord Detection :** In this approach, we have considered only the Major and Minor triads. However, in addition to these, there are other chord possibilities in popular music and future work would be targeted towards the detection of the Dominant 7^{th} category chords and extended chords as discussed in section 5.3. The chord detection research can be further extended to include knowledge of chord progressions based on the function of chords in their diatonic scale, which relates to the expected resolution of each chord within a key. That is, the analysis of chord progressions based on the "need" for a sounded chord to move from an unstable sound (dissonance) to a more final or stable sounding one (a consonance).

- **Rhythm Tracking :** The rhythm extraction technique employed in our current system does not perform very well for drumless music signals since the onset detector has been optimized to detect the onset of percussive events. Future effort will be aimed at extending the current work for music signals that do not contain drum sounds.

Appendix A

Publications

- A. Shenoy, R. Mohapatra, and Y. Wang. Key Determination of Acoustic Musical Signals. In *ICME*, 2004

  ► (Major contribution - conceptualized, designed and implemented the framework.)

- Y. Wang, M.Y. Kan, T.L. Nwe, A. Shenoy, Y. Jun. LyricAlly: Automatic Synchronization of Acoustic Musical Signals and Textual Lyrics. In proc. *ACM-MM*, 2004

  ► (Significant contribution - integrated Rhythm Detection into the system, one of the four core system components. Best student paper award)

- T.L. Nwe, A. Shenoy, and Y. Wang. Singing Voice Detection in Popular Music. In proc. *ACM-MM*, 2004

  ► (Significant contribution - pre-processing of the audio using musically designed Key filters, editorial comments.)

- N.C. Maddage, C. Xu, A. Shenoy, and Y. Wang. Semantic Region Detection in Acoustic Music Signals. To appear in proc. *PCM*, 2004

  ► (Minor contribution - editorial comments.)

Bibliography

[1] P. Aigrain. New applications of content processing of music. *JNMR*, 28(4):271–280, December 1999.

[2] P. E. Allen and R. B. Dannenberg. Tracking musical beats in real time. In *ICMC*, 1990.

[3] J. Backus. *The Acoustical Foundations of Music*. W.W. Norton and Company, December 1977. 2nd edition.

[4] M. A. Bartsch and G. H. Wakefield. To catch a chorus: Using chroma-based representations for audio thumbnailing. In *WASPAA*, 2001.

[5] J.P. Bello, G. Monti, and M. Sandler. Techniques for automatic music transcription. In *ISMIR*, 2000.

[6] F. Carreras, M. Leman, and M. Lesaffre. Automatic harmonic description of musical signals using schema-based chord decomposition. *JNMR*, 28(4):310–333, December 1999.

[7] A. T. Cemgil and B. Kappen. Monte carlo methods for tempo tracking and rhythm quantization. *JAIR*, 18:45–81, 2003.

[8] A.T. Cemgil, B. Kappen, P. Desain, and H. Honing. On tempo tracking: tempogram representation and kalman filtering. *JNMR*, 28(4), 2001.

[9] A. Copland. *What to Listen for in Music*. Penguin USA, March 1999.

[10] S. Dixon. Automatic extraction of tempo and beat from expressive performances. *JNMR*, 30(1), 2001.

[11] S. Dixon. On the analysis of musical expressions in audio signals. *SPIE*, 5021(2):122–132, 2003.

[12] G. Ewer. *Easy Music Theory*. Spring Day Music Publishers, 2002.

[13] T. Fujishima. Realtime chord recognition of musical sound: A system using common lisp music. In *ICMC*, 1999.

[14] M. Goto. A chorus-section detecting method for musical audio signals. In *ICASSP*, 2003.

[15] M. Goto. An audio-based real-time beat tracking system for music with or without drum-sounds. *JNMR*, 30(2):159–171, June 2001.

[16] M. Goto and Y. Muraoka. A beat tracking system for acoustic signals of music. In *ACM Multimedia*, pages 365–372, 1994.

[17] M. Goto and Y. Muraoka. Real-time beat tracking for drumless audio signals: Chord change detection for musical decisions. *Speech Communication*, 27(3-4):311–335, 1999.

[18] K. Hevner. Experimental studies of the elements of expression in music. *American Journal of Psychology*, 48:246–268, 1936.

[19] D. Huron. Perceptual and cognitive applications in music information retrieval. In *ISMIR*, 2000.

[20] D. Huron and R. Parncutt. An improved model of tonality perception incorporating pitch salience and echoic memory. *Psychomusicology*, 12:154–171, 1993.

[21] A. Klapuri. Automatic transcription of music. In *SMAC*, 2003.

[22] C. Krumhansl. *Cognitive Foundations of Musical Pitch*. Oxford University Press, Oxford, 1990.

[23] P. Lynn. *The Functions of Chords: For Pop, Jazz, and Modern Styles*. P & R Press, September 2002.

[24] MIT Media Laboratory Machine Listening Group. Mpeg-4 structured audio. http://sound.media.mit.edu/mpeg4.

[25] K.D. Martin, E.D. Scheirer, and B.L. Vercoe. Music content analysis through models of audition. In *ACM Multimedia Workshop on Content Processing of Music for Multimedia Applications*, 1998.

[26] Musicnotes.com. Commercial sheet music resource. http://www.musicnotes.com.

[27] K. Ng, R. Boyle, and D. Cooper. Automatic detection of tonality using note distribution. *JNMR*, 25(4), 1996.

[28] L.I. Ortiz-Berenguer and F.J. Casajus-Quiros. Polyphonic transcription using piano modeling for spectral pattern recognition. In *DAFX*, 2002.

[29] B. Pardo and W. Birmingham. Automated partitioning of tonal music. Technical Report CSE-TR-396-99, Electrical Engineering and Computer Science Department,University of Michigan, 1999.

[30] B. Pardo and W. Birmingham. Chordal analysis of tonal music. Technical Report CSE-TR-439-01, Electrical Engineering and Computer Science Department,University of Michigan, 2001.

[31] J. Pass. *Joe Pass Guitar Chords*. Mel Bay Publications, September 1986.

[32] R. D. Patterson. Spiral detection of periodicity and the spiral form of musical scale. *Psychology of Music*, 1986.

[33] J. Pickens. Key-specific shrinkage techniques for harmonic models. In *ISMIR*, 2003.

[34] J. Pickens and T. Crawford. Harmonic models for polyphonic music retrieval. In *CIKM*, 2002.

[35] J. Pickens, J. P.Bello, G. Monti, T. Crawford, M. Dovey, M. Sandler, and D. Byrd. Polyphonic score retrieval using polyphonic audio queries: A harmonic modeling approach. In *ISMIR*, 2002.

[36] M. D. Plumbley, S. A. Abdallah, J. P. Bello, M. E. Davies, G. Monti, and M. B. Sandler. Automatic music transcription and audio source separation. *Cybernetics and Systems*, 33(6):603–627, 2002.

[37] D.J.L. Povel. A model for the perception of tonal melodies. In *ICMAI*, 2002.

[38] C. Raphael. Automated rhythm transcription. In *ISMIR*, 2001.

[39] C. Raphael. Automatic transcription of piano music. In *ISMIR*, 2003.

[40] C. Raphael and J. Stoddard. Harmonic analysis with probabilistic graphical models. In *ISMIR*, 2003.

[41] E. D. Scheirer. Tempo and beat analysis of acoustic musical signals. *JASA*, 103(1):588–601, 1998.

[42] W.A. Sethares, R.D. Morris, and J.C. Sethares. Beat tracking of audio signals using low level audio features. *IEEE Trans. On Speech and Audio Processing (accepted for publication)*.

[43] Sheetmusicplus.com. commercial sheet music resource. http://www.sheetmusicplus.com.

[44] A. Sheh and D. Ellis. Chord segmentation and recognition using em-trained hidden markov models. In *ISMIR*, 2003.

[45] R. Shepard. Circularity in judgments of relative pitch. *JASA*, 36(12):2346–53, 1964.

[46] N.A. Smith and M.A. Schmuckler. Pitch-distributional effects on the perception of tonality. In *ICMPC*, 2000.

[47] B. Sollberger, R. Reber, and D. Eckstein. Musical chords as affective priming context in a word-evaluation task. *Music Perception*, 20(3):263–282, Spring 2003.

[48] D. Temperley. Improving the krumhansl-schmuckler key-finding algorithm. In *SMT*, 1999.

[49] D. Temperley. What's key for key? the krumhansl-schmuckler key-finding algorithm reconsidered. *Music Perception*, 17(1):65–100, 1999.

[50] B. Vercoe. Perceptually-based music pattern recognition and response. In *ICPCM*, 1994.

[51] G.H Wakefield. Mathematical representation of joint time-chroma distributions. In *SPIE*, 1999.

[52] Y. Wang, J. Tang, A. Ahmaniemi, and M. Vaalgamaa. Parametric vector quantization for coding percussive sounds in music. In *ICASSP*, 2003.

[53] Y. Zhu and M. Kankanhalli. Music scale modeling for melody matching. In *ACM-MM*, 2003.

[54] Y. Zhu, M. Kankanhalli, and S. Gao. A method for solmization of melody. In *ICME*, 2004.