PhD Dissertation

International Doctorate School in Information and Communication Technologies

DISI - University of Trento

MUSIC SIGNAL PROCESSING FOR AUTOMATIC EXTRACTION OF HARMONIC AND RHYTHMIC INFORMATION

Maksim Khadkevich

Advisor:

Prof. Maurizio Omologo

Università degli Studi di Trento

December 2011

Abstract

This thesis is concerned with the problem of automatic extraction of harmonic and rhythmic information from music audio signals using statistical framework and advanced signal processing methods.

Among different research directions, automatic extraction of chords and key has always been of a great interest to Music Information Retrieval (MIR) community. Chord progressions and key information can serve as a robust mid-level representation for a variety of MIR tasks. We propose statistical approaches to automatic extraction of chord progressions using Hidden Markov Models (HMM) based framework. General ideas we rely on have already proved to be effective in speech recognition. We propose novel probabilistic approaches that include acoustic modeling layer and language modeling layer. We investigate the usage of standard N-grams and Factored Language Models (FLM) for automatic chord recognition. Another central topic of this work is the feature extraction techniques. We develop a set of new features that belong to chroma family. A set of novel chroma features that is based on the application of Pseudo-Quadrature Mirror Filter (PQMF) bank is introduced. We show the advantage of using Time-Frequency Reassignment (TFR) technique to derive better acoustic features.

Tempo estimation and beat structure extraction are amongst the most challenging tasks in MIR community. We develop a novel method for beat/downbeat estimation from audio. It is based on the same statistical approach that consists of two hierarchical levels: acoustic modeling and beat sequence modeling. We propose the definition of a very specific beat duration model that exploits an HMM structure without self-transitions. A new feature set that utilizes the advantages of harmonic-impulsive component separation technique is introduced.

The proposed methods are compared to numerous state-of-the-art approaches by participation in the MIREX competition, which is the best impartial assessment of MIR systems nowadays.

Acknowlegements

I would like to give special thanks to a number of people for the completion of this dissertation. First and foremost, extreme gratitude to my advisor, Prof. Maurizio Omologo. He has taught me every step of the way how research activities are performed. Without him this would have never been completed. I would like to say thanks to my SHINE colleagues for the valuable discussions and friendly atmosphere.

I was lucky enough to have the opportunity to visit Audio, Acoustics and Waves Group at Télécom ParisTech, Signal and Image Processing Department. The group has been a source of good advice and collaboration. During the three months in Paris, I was given the chance to learn much thanks to discussion and help from Prof. Richard and Dr. Thomas Fillon.

Lastly, I would like to express special gratitude to my family for all their encouragement and love, to my friends who supported me in all my pursuits. And most of all, I would like to thank my other half, thank you for your never ending love and faithful support.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis deals with automatic extraction of harmonic and rhythmic information from raw audio. This chapter makes an introduction to MIR, formulates the motivation, sets goals and describes the contributions.

1.1 Content-based music information retrieval

Recent advances in digital media have allowed for extensive wide-spread growth of musical collections. Existing storage capacities allow for having huge collections of media on portable media devices. There is a continuous transformation of the way we listen to music. Going back to the end of the 20-th century, we could observe radio broadcasting and music record stores to be the major ways of music consumption. Nowadays, drastic popularity of social networking has lead to the creation of web communities, changing the way of music dissemination. Music recommendation services, such as Last.fm[1] have gained huge popularity and proposed new facilities to access media data based on your personal preferences.

To this end, the need for effective search in large media databases is becoming critical. Developing techniques for accessing content and browsing in huge music archives has become an emerging area of research. High demand for such techniques has lead to establishing and evolving Music Information Retrieval (MIR) community, which include academic research institutions, as well as industrial research companies. Music Information Retrieval is an interdisciplinary science that addresses extraction of meaningful information from music data. It involves musicology, signal processing, machine learning and other disciplines.

In spite of growing research activities in MIR, nowadays, the most common way of media search is accomplished through textual metadata. Lots of music download services are based on the search by artist, album, song name. However, a number of content-based search engines,

[1] http://www.last.fm

such as Shazam[2] and SoundHound[3] have become available, introducing essentially novel approaches to music retrieval. Content-based concept is based on the principle of processing the content of a given audio document and extracting the necessary information from it.

Shazam provides music identification service that is based on acoustic fingerprinting [1]. A fingerprint of each audio file from a huge music database is stored in an indexed database. A query audio file is subjected to the same analysis and the extracted fingerprint is matched against a large set of fingerprints from the database. A list of possible candidate songs from the database is evaluated for the correctness of the match. For robust identification in noisy environment, spectrogram peaks are used as feature set. A set of time-frequency points with the highest energy in their neighboring region is extracted and constructed constellation map is indexed. The robustness of the approach is proved by the fact that noise and distortions usually do not change the temporal layout of the spectrogram peaks. Shazam is considered to be an effective tool to search for exact content match. However, slight modifications in the song arrangement make identification impossible. For example, search query using a remix version for a given song would fail.

However, the solution of the above-mentioned search problem, when there is no exact match in spectral peak distribution, but high similarity in the harmonic content is proposed by SoundHound. Apart from the functionality provided by Shazam, the algorithm is so advanced that it can confidently recognize a song from your own singing and/or humming. On the other hand, sometimes, the system is not capable of exact matching and can provide a remixed version of a query song instead of the original as the final result.

Shazam and SoundHound are the solutions developed mainly for the mobile phone users. However, there is the need for such tools on desktop computers. A possible scenario could be the following: having a huge amount of untagged music data, organize a collection, where songs are sorted according to a certain criteria, e. g. artist/album, style. A solution is proposed by MusicBrainz[4] project. The project is maintained by open community that collects, and makes available to the public music metadata in the form of a relational database. The database of MusicBrainz contains information about artists, track titles, the length of each track, and other metadata. Recorded works can additionally store an acoustic fingerprint for each track. This provides the facility for automatic content-based identification and subsequent tagging.

There are thousands of other possible applications of the MIR technologies. For all of them, effective and robust algorithms for feature extraction play an essential role.

[2] http://www.shazam.com

[3] http://www.soundhound.com/

[4] http://www.musicbrainz.org

1.1.1 High-level music descriptors

One of the largest research areas of MIR is the extraction of high-level music descriptors, or attributes. The most important and informative attributes include harmony, rhythm, melody, instrumentation and others. Effective methods for extraction such descriptors is the necessary condition for developing robust and effective music information retrieval systems. A fundamental approach to the classification of musical facets was proposed by Orio [2].

In the following sections a short description of the most important high-level music descriptors is provided.

Onset structure

An important characteristic of any musical excerpt is the onset structure. Onset information can be useful for the analysis of temporal structure such as tempo and meter. Music classification and music fingerprinting are the tasks where onset information could also be of great use [3]. The notion of onset leads to many definitions: a sudden burst of energy, a change in the short-time spectrum of the signal or in the statistical properties. The onset of the note is a single instant chosen to mark the temporally extended transient. In most cases, it will coincide with the start of the transient, or the earliest time at which the transient can be reliably detected [4]. Onsets can be divided into two classes, "soft" and "hard". A hard onset is characterized by a sudden energy change. A soft onset is usually represented by slow changes in the spectral energy. The most straightforward methods for hard onset detection are based on the analysis of energy-based features. Soft onsets are considered to be much more difficult to detect and usually involves spectral analysis methods. Noise and oscillations associated with frequency and amplitude modulation make the task of onset structure extraction challenging.

Rhythmic structure

Rhythmic structure of music plays an important role in MIR-related tasks. It is primarily represented by tempo, beat and meter structure. For example, knowing beat structure allows one to extract musically meaningful beat-synchronous features, instead of performing frame-by-frame analysis. It can be of great benefit to manage the tasks of music structure extraction or cover song identification. In these tasks dealing with beat-normalized time axis is usually much more convenient, since a tempo-invariant representation is utilized.

Rhythmic structure is strongly related to the notion of meter. Meter can be characterized by a hierarchical structure that comprises several levels [5]. Perceptually, the most important level is the tactum-level, which is also referred to as the beat-level. It usually corresponds to the period of foot-tapping. Bar-level structure is another important information, which is characterized by the number of tactum-level events within one musical measure. Bar-level structure is also

named as time signature and can be expressed in the form of a fractional number, e.g. 3/4, 2/4, 6/8. It gives information on the organization of strong (bar-level) and soft (tactum-level) events along the time axis. An example of hierarchical rhythmic structure is presented in Figure 1.1

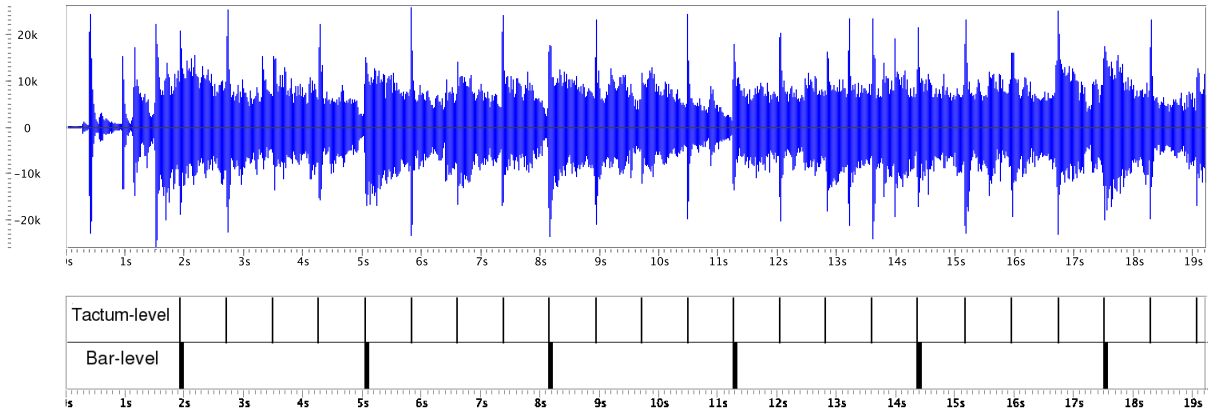


Figure 1.1: An example of hierarchical rhythmic structure for the beginning of George Michael's "Careless Wisper"

Nowadays, beat detection is one of the most challenging tasks in the MIR community. While processing modern rock and pop songs with rich percussive part and stable rhythm is a nearly solved problem, dealing with non-percussive music with soft note onsets and time-varying tempo, that is characteristic of classical music, is still a challenge.

Melody

Melody is amongst the most important high-level descriptors that describe the contents of music signals. Melody extraction is highly related to the general task of pitch detection and tracking that has been extensively addressed in other research areas, such as speech signal processing. However, the task of melody extraction does not only mean estimation of the fundamental frequency, but also the subsequent quantization using musical scale to produce a score-like representation. As in the case of single speaker in speech processing, melody detection in the case of monophonic signals is nearly a solved problem. However, dealing with multi-instrument signals with the number of fundamental frequencies at a given time instant greater than one is still a challenge. This problem becomes even harder, if accompaniment instruments have rich spectral representation with harmonics containing significant part of spectral energy. There are some problems one can come across when extracting melody. Some performances may contain vibrato parts, which can lead to a sequence of notes in the final transcription, while the original score notation contain just a single note. Another case that is hard to manage is glissando. In this case, rapidly changing pitch can also be transcribed as a sequence of notes.

Melody is considered to be the attribute that captures the most significant information about

song. A song that needs to be recalled can be easily represented by singing or humming the melody, since in most cases, melody is the attribute that distinguishes one piece from another.

Harmonic structure

Key and chords are the two attributes that describe tonal and harmonic properties of a musical piece. Harmony denotes a combination of simultaneously or progressively sounding notes, forming chords and their progressions. Among all existing musical styles, western tonal music, which is one of the most popular nowadays, is known for its strong basement on harmony. Harmonic structure can be used for the purposes of content-based indexing since it is derived from the mood, style and genre of musical composition. Harmonic structure can be described in terms of chord sequences. A chord can be introduced as a number of notes sounding simultaneously, or in a certain order between two time instants, known as chord boundaries. Therefore, the task of chord transcription includes chord type classification and precise boundary detection.

Harmony together with such features as tempo, rhythm, melody extracted from a raw waveform can be widely used for context-based indexing, retrieval, and navigating through large collections of audio data.

1.1.2 MIR applications

Extracting high-level information, such as rhythm, harmony, key, melody has become a challenge. We have entered an era of complex content-based search and retrieval systems [6]. A number of use cases, where recently developed content-based methods were successfully applied in MIR applications are addressed in this section.

Automatic music transcription and lead sheet generation

Similarly to automatic speech recognition, automatic music transcription has a lot of challenging tasks. For example, distinguishing musical instruments in a polyphonic piece of audio can be more or less easily done by human being. Meanwhile solving this problem automatically needs a lot of research effort. Actually the most daunting problem is the transcription of polyphonic piece of music in terms of notes, which implies producing score notation for each instrument. The subtask of this problem, which deals with the extraction of harmonic properties of audio signal, is chord recognition. Another challenging subtask is the extraction of hierarchical rhythmic structure [5].

Recently, systems that are capable of comprehensive music transcription have become available. For example, Weil et al. [7] proposed a lead sheet generation system that aggregates high-level music descriptors. Tempo, time signature, melody, chords, and key are extracted in separate modules that can interact with each other. Rendered lead sheets contain the melody

and the corresponding chord progression in a beat-synchronous fashion. State-of-the-art performance in each constituent module is achieved, which allows for obtaining transcription results close to musician expectations.

Accompaniment based on chord extraction

While the application area of tempo and beat descriptors is mainly indexing and segmentation, the information on chord progressions covers more practical aspects. Opportunity to automatically extract harmonic structure can be of great use to musicologists, who perform harmonic analysis over large collections of audio data, or just to amateur musicians. A great interest in chords can be indicated by the number of websites containing chord databases for existing songs. Archives containing chord transcriptions are becoming more and more popular. An easy way to accompany a singer is to play the chords extracted from the performed song, which can be extracted manually by expert musicians, or in automatic fashion. For the moment, the content is generated by users manually in a time-consuming manner. The quality of the data highly depends on the user expertise and background in music. That is why online chord databases sometimes contain not reliable transcriptions. At the same time, modern advanced automatic chord extraction systems do not allow to produce 100% correct labels. The best system in the MIREX 2011 competition performed at 83% recognition rate.

The compromise between time-consuming manual labeling and the quality of automatic chord transcription can be achieved in semi-automatic mode. In the first step, preliminary labels are obtained by running automatic chord extraction system. In the final step, a number of trained musicians work together on error correction and quality check.

Automatic accompaniment generation for vocal melody and automatic song creation

Melody and harmony are considered to be the backbone of a song. The process of song creation for many song writers often starts with the idea about melody [8]. In this approach developing chord progression and accompaniment patterns are the necessary steps to produce the final version. Usually, professional musicians with the knowledge of musical structure and harmony manage the whole process of song production. However, people with poor background in music theory are not able to participate in such an amusing and creative activity. Recent advances in MIR have allowed musically-untrained individuals to work on music creation. An example of a machine-learning-based system that takes a melody as an input and generates appropriate chords was presented by Simon et al. [8].

Another interesting use case of song generation was proposed by Fukayama et al. [9]. Automatic song generation web-service was developed in the context of Orpheus[5] project. The

[5]http://ngs.hil.t.u-tokyo.ac.jp/∼orpheus/cgi-bin/

system requires only song lyrics as an input data. A user can also set up music genre, voice, tempo and other parameters. Then the system performs text analysis, melody and harmony generation, and produces score notation containing lyrics along with the resultant audio file.

Recommender systems

Extreme growth of online music collections and advances in digital multimedia have allowed us to start listening music just with a click of a button. In spite of the easy access to large web archives, discovering new music according to our personal preferences is a hard problem. This caused a variety of music recommender systems to come into existence. There are several approaches to music recommendation.

Pandora[6] is one of the most popular music recommendation systems. It is based on the Music Genome Project[7]. Each track in the database is annotated with 400 different attributes. Annotating is performed in a time-consuming manual fashion by professional musicians, which makes the growth of the database dependent on the human resources.

Music recommendation system proposed by Last.fm[8] is based on a different approach. They have developed social recommenders, also known as collaborative filters. The statistics for music tracks ever listened by a particular user forms the basis of the recommendation engine. Each user is proposed to install an optional plug-in that monitors media player software and builds a profile of his or her music preferences. Having a large database of user profiles, the system finds users whose listening history is similar and makes suggestions.

Mufin[9] is a music recommendation service that is purely content-based. It analyzes the fundamental properties of a song. The recommendation is based on the similarity of the content. The algorithm analyzes 40 characteristics of each song, including tempo, sound density, and variety of other factors.

Other use cases

The number of possible use cases, where content-based MIR algorithms are successfully applied is not limited to the above-mentioned applications. Artist identification, copyright infringement detection and protection, instrument separation, performance alignment, plagiarism detection, composer identification are amongst the most challenging MIR tasks being addressed recently.

[6] http://www.pandora.com
[7] http://www.pandora.com/mgp.shtml
[8] http://www.last.fm
[9] http://www.mufin.com

1.1.3 Music Information Retrieval Evaluation eXchange

Progressive and continuous evolving of MIR systems that we can observe nowadays is boosted by the existence of Music Information Retrieval Evaluation eXchange (MIREX) [10], the most influential, community-based evaluation framework. Direct comparison of MIR systems aimed at solving a specific problem is sometimes impossible due to many factors. Performance of a given system can be obtained on different datasets, using different evaluation metrics. Establishing common rules of MIR system assessment has become a necessity and caused establishing and gradual development of the MIREX framework.

Audio Classification (Train/Test) Tasks, incorporating:
Audio US Pop Genre Classification
Audio Latin Genre Classification
Audio Music Mood Classification
Audio Classical Composer Identification
Audio Cover Song Identification
Audio Tag Classification
Audio Music Similarity and Retrieval
Symbolic Melodic Similarity
Audio Onset Detection
Audio Key Detection
Real-time Audio to Score Alignment (a.k.a Score Following)
Query by Singing/Humming
Audio Melody Extraction
Multiple Fundamental Frequency Estimation & Tracking
Audio Chord Estimation
Query by Tapping
Audio Beat Tracking
Structural Segmentation
Audio Tempo Estimation

Table 1.1: MIREX 2011 tasks.

MIREX is coordinated by the International Music Information Retrieval Systems Evaluation Laboratory (IMIRSEL) [11] at the University of Illinois at Urbana-Champaign. The target of IMIRSEL is to create the necessary infrastructure for the evaluation of many different MIR systems. The necessary condition for any kind of MIR evaluations is the music data collections with the corresponding metadata information. Due to different copyright issues, sometimes it is not possible to provide public access to the data. Another reason not to give an access to the

test datasets for the participants is to prevent from model parameter over-fitting, which is an important aspect of objective evaluation. To this end, participants should deliver their systems to the MIREX team to execute using recently developed The Networked Environment for Music Analysis (NEMA) framework [12]. The framework facilitates valid statistical comparisons between techniques, avoiding the above-described problems by carrying out experiments that are both carefully constructed and exactly repeatable.

The set of different tasks is defined by the community. Anyone is free to propose a new task, describing the evaluation metrics, and, if necessary, provide a dataset. Then, the task is discussed on the wiki-pages by all potential participants, different approaches for the evaluation are taken into consideration, and the final description and rules becomes available.

Starting from the MIREX of 2005 year, a lot of different tasks have been proposed. Table 1.1 contains a list of tasks for the MIREX of 2011 year.

1.2 Motivation

Fast development of hi-tech industry allowed people having hours of digital audio recordings in their pockets. It caused high demand for content-based search and retrieval, known as music recommendation. Due to the extreme growth of digital music collection, effective and robust content-based indexing and retrieval methods have become an emerging area of research. It boosted the demand for tools that can perform accurate extraction of high-level descriptors. Chords, key, beat structure and tempo are among the most relevant descriptive attributes of music information.

Given the great demand for tools that are able to perform content-based analysis, higher level aspects of musical structure, such as harmony and rhythm are given attention, and we contribute exploring these areas.

1.3 Goals

This thesis encompasses a variety of research activities aimed mainly at the extraction of harmonic and rhythmic descriptors. The main focus is concentrated on the developing computational algorithms and effective feature extraction methods for the transcription of chord sequences and beat structure.

The objectives of the work include the following aspects:

1. Analyze state-of-the art approaches for chord detection and beat structure extraction.
2. Develop robust feature sets that capture essential information from audio for a given task.
3. Design and develop probabilistic frameworks for automatic chord recognition and beat/downbeat extraction.
4. Perform large-scale evaluations and describe the behavior of the developed systems varying different configuration parameters.
5. Participate in the MIREX competition to demonstrate the competitiveness of the proposed approaches.

1.4 Contributions

The thesis contribute in the following areas:

1. A new feature set for chord recognition that outperforms standard chroma feature has been proposed. It is based on the Time-Frequency Reassignment technique and incorporates harmonic-impulsive component separation.

2. A two-level probabilistic framework for chord recognition has been introduced. It is based on a novel approach that includes acoustic modeling layer and language modeling layer.

3. The usage of standard N-grams and Factored Language Models for automatic chord recognition has been addressed. Experiments with different back-off strategies for Factored Language Models have been carried out.

4. The performance of the proposed chord recognition system has been investigated using large-scale parameter optimization.

5. A new feature set for beat/downbeat detection has been proposed. It is based on the harmonic and impulsive part of the Time-Frequency Reassigned spectrogram.

6. A novel probabilistic approach to beat/downbeat detection has been developed. The definition of a very specific beat duration model that exploits an HMM structure without self-transitions has been introduced.

7. All the described techniques have been implemented and submitted to the MIREX [13] competition. Our chord recognition system showed the best result in the 2011 year contest, while our beat/downbeat estimation system was at the top of the list for the MCK dataset.

Part I

Chord recognition

Chapter 2

Background

This chapter is concerned with the background information on chord extraction. Among all existing musical styles, western tonal music, which is one of the most popular nowadays, is known for its strong relationship to harmony.

Harmonic structure can be used for the purposes of content-based indexing and retrieval since it is correlated to the mood, style and genre of musical composition. It has been successfully used for audio emotion classification [14], cover song identification [15], audio key estimation [16]. Chord sequence can serve as a robust mid-level representation for a variety of MIR tasks. Among different research directions automatic chord recognition has always been of a great interest to MIR community.

During the past few decades several approaches to chord recognition were developed. They can be classified into template matching[17, 18], machine learning [19–21] and hybrid approaches [22, 23]. The majority of the current state-of-the-art machine learning approaches are based on Hidden Markov Models [24], [25], [26], Dynamic Bayesian Networks (DBN) [27] and Support Vector Machines (SVM) [28]. Submissions based on the above cited approaches were among the top-ranked results in the MIREX competitions.

Section 2.1 introduces general information on feature vector selection and extraction techniques. In Sections 2.2 – 2.3 different approaches to automatic chord recognition are presented.

2.1 Feature extraction for chord recognition

2.1.1 Feature selection

As in the case of speech recognition, one of the most critical issues in chord recognition is the choice of the acoustic feature set to use in order to represent the waveform in a compact way.

Chromagram has been the most successfully used feature for the chord recognition task. It consists of a sequence of chroma vectors. Each chroma vector, also called Pitch Class Profile

(PCP), describes harmonic content of a given frame. The amount of energy for each pitch class is described by one component in a chroma vector. Since a chord consists of a number of tones and can be uniquely determined by their positions, chroma vector can be effectively used for the chord representation.

Fujishima was the first one who used the chroma feature [29] for chord extraction from audio. The most common way of calculating chromagram is to transform the signal from the time domain to the frequency domain with the help of short-time Fourier transform (STFT) or constant-Q transform and subsequent energy mapping of spectral bins to chroma bins [19–21, 30–32].

An alternative way to extract chroma was proposed by Müller [33]. The analyzed signal is passed through a multirate IIR filter bank. In the first step, STMSP (Short-Time Mean-Square Power) features that measure the local energy content of each filter output are extracted. Large amount of energy indicate the presence of musical notes whose frequencies correspond to the passband of a given filter. In the next step, chroma-based audio representation is obtained from STMSP by summing energies that correspond to the subbands of the same pitch class.

Much attention has been put to the problem of higher harmonics and their impact on the chroma vector. Several approaches proposed performing some sort of harmonic analysis in order to reveal the presence of higher harmonic components [34–36]. All these approaches are based on a frame-by-frame spectral analysis that is aimed at finding all the pitches that occur in the given time instant.

In the approach of Mauch and Dixon [34] an approximate note transcription procedure was applied before calculation of wrapped chromagram. Experimental results showed an increase in performance of 1%. However, their technique proved to be more advantageous when considering "difficult" chords.

Ueda et al. [26] showed the importance of harmonic filtering step for feature extraction. Before extracting feature vectors, a harmonic/percussive separation is performed in order to remove impulsive components and noise. The system based on this approach showed the best result in the MIREX 2008 competition. Another important issue the authors addressed in this paper is the usage of dynamic delta-features.

There were some attempts to use features derived from standard chroma vector using an additional transform operation. Lee and Slaney [20] used tonal centroid as an alternative to chroma vectors. In their experiments on the first two Beatles albums, as well as on two classical pieces of Bach and Haydn tonal centroid showed to outperform chroma features. Another example of feature set obtained from chroma is presented in the approach of Ueda et al. [26]. They used FFT of the chroma vectors as feature set for chord recognition system and showed the advantage of this transform in terms of recognition rate.

2.1.2 Beat-synchronous features

Recently, several approaches that exploit mutual dependency between harmonic progressions and rhythmic structure have been proposed [21], [37], [38]. Beat-synchronous chroma features are used instead of frame-by-frame chroma vectors [37], [28]. Since western music is highly structural in terms of rhythm and harmony, the basic idea that chord boundaries occur on the beat positions is exploited.

Papadopoulos and Peeters [21] proposed a system that performs simultaneous estimation of chord progressions and downbeats from audio. They paid a lot of attention to possible interaction of the metrical structure and the harmonic information of a piece of music. They proposed a specific topology of HMMs that allows for modeling chords dependency on metrical structure. Thus, their system is capable of recognizing chord progressions and downbeat positions at the same time. The model was evaluated on a dataset of 66 Beatles songs and proved to improve both the estimation of the chord progression and the downbeat positions.

Bello and Pickens [37] used a similar approach. The evaluation of their system showed a significant increase in performance (about 8%) when using beat-synchronous chroma features as opposed to frame-by-frame approach.

However, beat-synchronous features have some weak sides. Since the quality of beat-level segmentation depends highly on the beat extraction approach, some beat location errors can lead to incorrect segmentation.

2.1.3 Tuning

In the stage of feature extraction for chord recognition and key estimation, a lot of attention has been paid to tuning issues [18, 30, 31]. The necessity of tuning appears when audio was recorded from instruments that were not properly tuned in terms of semitone scale. They can be well-tuned relatively to each other, but, for example, "A4" note is played not at conventional 440 Hz but at 447Hz. This mis-tuning can lead to worse feature extraction and, as a result, less efficient or incorrect classification. Several approaches to circumvent the problem have been developed.

Harte and Sandler [18] suggested using 36 dimensional chroma vectors. The frequency resolution in this case is one-third of a semitone. After the peak-picking stage and computing a histogram of chroma peaks over the entire piece of music they find mis-tuning deviation. And prior to calculating 12-bin conventional chromagram they accurately locate boundaries between semitones. The resulting 12-bin semitone-quantized chromagram is then compared with a set of predefined chord templates. They defined 4 chord types - major, minor, diminished and augmented for each pitch class (total 48 chord templates). Two full albums of the Beatles were used for evaluation. The average frame-level accuracy was 62.4%.

Peeters [31, 32] tested a set of candidate tunings, i.e. the quarter-tone below and the quarter-tone above "A4" note. For each possible tuning the amount of energy in the spectrum is estimated. After defining the global tuning center, the signal is resampled so that it becomes tuned to 440Hz.

Mauch et al. [30] used a quite similar approach: after computing 36-bin chromagram they pick one of three possible sets of 12-bin chromagram, relying on the maximum energy inside candidate bins (e. g. {1, 4, 7... 34 }).

2.2 Template-matching techniques

Template matching techniques are based on the idea of introducing a set of templates for each chord type. The template configurations are derived either heuristically or using some knowledge from music theory. In the classification step, extracted feature vectors are matched against all possible templates. The template that produces the highest correlation is used to generate chord label for a given vector.

A most trivial example would be the definition of a binary 12-dimensional chord template mask, where pitch classes that correspond to the constituent notes of a given chord are set to ones, while the other pattern components are set to zeros. A binary template T is defined as

$$T = [Z_C, Z_{C\#}, Z_D, Z_{D\#}, Z_E, Z_F, Z_{F\#}, Z_G, Z_{G\#}, Z_A, Z_{A\#}, Z_B] \tag{2.1}$$

where Z_p denotes the mask value that corresponds to the pitch class p. For example, binary masks for C major and D minor chords would take the following form:

$$\begin{aligned} T_{C:maj} &= [1,0,0,0,1,0,0,1,0,0,0,0] \\ T_{D:min} &= [0,0,1,0,0,1,0,0,0,1,0,0] \end{aligned} \tag{2.2}$$

Fujishima [29] proposed a real-time chord recognition system, describing extraction of 12-dimensional chroma vectors from the Discrete Fourier Transform (DFT) of the audio signal and introducing numerical pattern matching method using built-in chord-type templates to determine the most likely root and chord type. He introduced feature vector smoothing over time and "chord change sensing". The system was tested on real audio and showed 94% accuracy for the opening theme of Smetana's Moldau.

Similarly, Harte and Sandler [18] applied binary masks to generate templates for four different chord classes: major, minor, diminished and augmented. Their vocabulary consisted of 48 different chords. Evaluation was performed on the first two Beatles albums, "Please, Please Me" and "Beatles For Sale". The average frame-level accuracy they achieved was 62.4%.

Papadopoulos and Peeters [39] used more sophisticated chord templates that take into account higher harmonics of pitch notes. The ideas they rely on are based on the extension of

PCP, Harmonic Pitch Class Profile (HPCP) that was used by Gomez [40] for key detection. In the approach presented in [39], amplitude values of four – six higher harmonics contribute to chord templates. An empirical decay factor for higher harmonics amplitude is set to 0.6, so that the amplitude of a h-th harmonic is set to 0.6^h, where $h = 0$ corresponds to the fundamental. Evaluations on the 110 songs of Beatles showed that considering six higher harmonics in chord templates brings about 5% relative improvement.

Oudre et al. [41] proposed a template-based method for chord recognition. They investigate different chord models taking into account one or more higher harmonics. As in the above-mentioned approaches, the detected chord over a frame is the one minimizing a measure of fit between a rescaled chroma vector and the chord templates. An interesting investigation they carried out is the influence of different measures of fit between the chroma features and the chord templates. In order to take into account the time-persistence, they performed a post-processing filtering over the recognition criteria, which quickly smoothes the results and corrects random errors. Their system was evaluated on the 13 Beatles albums. The experiments showed that chord template configurations with one and four harmonics showed better results than those with six harmonics. They discovered that the most robust and effective measure of fits in their approach are the Kullback-Leibler divergence and the Euclidean distance.

A fast and efficient template-based chord recognition method was suggested in [17]. The chord is determined by minimizing a measure of fit between the chromagram frame and the chord templates. This system proved the fact that template-based approaches can be as effective as probabilistic frameworks.

2.3 Machine learning techniques

HMM-based aproaches

Sheh and Ellis [19] proposed a statistical learning method for chord recognition. The Expectation-Maximization (EM) algorithm was used to train Hidden Markov Models, meanwhile chords were treated as hidden states. Their approach involves statistical information about chord progressions – state transitions are identical to chord transitions. The optimal state path is found using the Viterbi algorithm. They achieved accuracy of 23% in the chord recognition task and 75% in the forced-alignment task, which is not longer state of the art. But their work made substantial contributions in several aspects. They applied much of the speech recognition framework with minimal modification. They draw an analogy between the sequence of discrete chord symbols used to describe a piece of music, and the word sequence used to describe recorded speech. It was shown that the chromagram is superior to Mel-frequency cepstral coefficients (MFCCs).

Bello and Pickens [37] proposed a probabilistic model that is partially based on music theoretical considerations. As opposed to Sheh and Ellis [19], who used random initialization of mean vector and diagonal covariance matrix in Gaussian distributions, they propose initialize these values according to music theory. In order to take into consideration correlation between chroma elements, they introduce full covariance matrix. They claim that pitches which comprise the triad are more correlated than pitches which do not belong to the triad. These dependencies are introduced when initializing covariance matrix. They propose selective model training using the standard Expectation Maximization (EM) algorithm for HMM parameter estimation as introduced in [42]. The observation vector parameters are not re-estimated in the training phase. The only parameters that are updated using EM algorithm is the chord transition matrix and initial distributions. The experiments were conducted using the first two Beatles albums, "Please, Please Me" and "Beatles For Sale". The performance of their system proved to be significantly higher when using selective model training (75%), if compared to the system configuration, where all parameters are re-estimated in the training phase (42%).

In Western tonal music, key and chord progression are the two artifacts that are highly dependent on each other. Some approaches exploit this mutual dependency [20],[43]. The advantage of such systems is the possibility of concurrent estimation of key and chord progression, which is achieved by means of building key-dependent HMMs.

Lee and Slaney [20] described a chord recognition system that used symbolic data, taken from MIDI[1] files, to train HMMs. This allowed them to avoid a time consuming task of human annotation of chord names and boundaries. At the same time, they synthesized audio from the same symbolic files and extracted feature vectors. They build a key-dependent HMMs, where chord transition probabilities are influenced by a given key. During the Viterbi decoding [42] the HMM with the highest log-likelihood determines the global key and is used to derive chord progression.

Hybrid approaches

Yoshioka et al. [22] presented an automatic chord transcription system, which is based on generating hypotheses about tuples of chord symbols and chord boundaries, and further evaluating the hypotheses, taking into account three criteria: acoustic features, chord progression patterns, and bass sounds. Thus, they first performed beat-analysis on raw audio to extract downbeat positions of a piece of music. Then, the most probable hypothesis about a chord sequence and the key were searched. Finally, the obtained most plausible hypothesis is produced as an output. A conventional 12-dimensional chroma feature is used as feature set. Pre-defined chord progression patterns reduce the ambiguities of chord symbol identification results. They evaluated their

[1] http://www.midi.org

system on one-minute excerpts from seven popular songs, and achieved 77% average accuracy.

This approach was further developed by Sumi et al. [23]. They mainly focused on the interrelationships among musical elements and made an attempt to efficiently integrate information about bass lines into probabilistic chord recognition framework. Their framework made it possible to deal with multiple musical elements uniformly and integrate statistical information obtained from music recordings. They particularly exploited the mutual dependency between chord sequences and bass lines in order to improve the accuracy of chord recognition. For pruning the search space, they define the hypothesis reliability as the weighted sum of three probabilities: the likelihood of Gaussian Mixture Models for the observed features, the joint probability of chord and bass pitch, and the chord transition N-gram probability. Evaluation on 150 songs from twelve Beatles albums showed the average frame-rate accuracy of 73.4%.

Some approaches used structural segmentation information for enhancing chord recognition by combining information from different occurrences of the same segment type for chroma calculation [44].

In [27], a 6-layered dynamic Bayesian network was suggested. In this network four hidden source layers jointly model key, metric position, bass pitch class and chord. The two observed layers model bass and treble content of the signal. This approach shows an example of how simultaneous estimation of beats, bass and key can contribute to the chord recognition rate.

Ni et al. [45] proposed a system for simultaneous estimation of chords, key, and bass notes. As opposed to the approach of Mauch [46], where some expert knowledge is used to set up system parameters, it is fully based on the machine learning approach, where all the parameters are estimated from training data.

Chord progression statistics

Incorporating statistical information on chord progressions into a chord recognition system is an important issue. It has been addressed in several works through different techniques. Mauch and Dixon [30] used one of the simplest forms of N-grams – the bigram language model. In the approaches of Papadopoulos and Peeters, Lee and Slaney [20, 21] chord sequence modeling is introduced through state transition probabilities in HMM. In their case "language model" is a part of HMM and is derived from the Markov assumption, where chord probability is defined by only one predecessor. A large study on the modeling of chord sequences by probabilistic N-grams was performed by Scholz et al. [47]. Unal et al. [48] used perplexity-based scoring to test the likelihoods of possible transcription sequences.

Chapter 3

Feature extraction

In this chapter, a new feature set for extracting harmonic information from audio content is introduced. The proposed features belong to the chroma family, which has always been a common and well-established feature set for chord recognition. When performing feature extraction, signal in the given analysis frame is assumed to be stationary and it is also assumed that no note transitions occur inside it. However, standard chroma extraction approaches, which are based on Short-Time Fourier Transform (STFT) or Constant-Q transform, require frame size to be long enough to provide reasonable frequency resolution. Transients and noise may cause energy assignment to some frequencies that do not occur in the signal. In this thesis, we investigate on alternative solutions to feature vector extraction for chord recognition. Along with the description of traditional approaches to chroma extraction, we propose two novel methods that are based on PQMF filter bank and Time-Frequency Reassignment respectively, and provide their comparative characteristics.

3.1 Introduction to chroma features

Feature extraction is an important step in the majority of MIR tasks. It allows for representing a waveform in a compact way, capturing the desired characteristics of the analyzed signal for further processing. In chord recognition domain, chroma has always been almost unique feature. One of the reasons, why chroma performs well, is the strong connection between the physical meaning of chroma vector components and music theory.

Generally, chroma feature extraction consists of the following steps. At first, audio signal is downsampled and converted to the frequency domain by means of Short-Time Fourier Transform (STFT) or Constant-Q transform applying a window function with a given overlapping factor.

After applying STFT, the power spectrum is mapped to the chroma domain, as

$$n(k) = 12\log_2\left(\frac{f_k}{f_{ref}}\right) + 69, n \in \Re^+, \tag{3.1}$$

where f_{ref} denotes the reference frequency of "A4" tone, while f_k and n are the frequencies of Fourier transform and the semitone bin scale index, respectively. Usually, different approaches consider the range of frequencies between 100-200 Hz and 1-2 kHz, mainly because in this range the energy of the harmonic frequencies is stronger than non-harmonic frequencies of the semitones. In order to reduce transients and noise, similarly to Peeters [32] and Mauch et al. [30], smoothing over time using median filtering is applied. After filtering semitone bins are mapped to pitch classes, as follows:

$$c(n) = \mathrm{mod}(n, 12) \tag{3.2}$$

A sequence of conventional 12-dimensional chroma vectors, known as chromagram is used as acoustic feature set. Each element of chroma vector corresponds to the energy of one of the 12 pitch classes.

3.2 Tuning

An important parameter in Equation (3.1) that greatly influences the quality of chroma features is the reference frequency of "A4" note f_k. The task of f_k extraction is known as audio recording mis-tuning estimation problem. In this section, we propose a method for f_k estimation that is based on the analysis of the spectral phase change.

In order to circumvent the problem of audio recording mis-tuning, a technique that was formerly developed for phase vocoder [49] is utilized to estimate the reference frequency. The proposed method allows for very precise and accurate frequency estimation of each sinusoid by performing the analysis of the degree of phase change. The block diagram of the proposed estimation scheme is depicted in figure 3.1.

The basic principle is to compute a second Fourier transform of the same signal, windowed by the same function shifted by D samples. Let $x[n]$ be a sequence of samples of the analyzed signal that contains some fundamental and harmonic components. Discrete Fourier Transform (DFT) is performed on the signal weighted by window function $w[n]$ as

$$X_w[t_0, k] = \sum\nolimits_{n=0}^{N-1} w[n]x[n + t_0]e^{-2\pi jnk/N} \tag{3.3}$$

where k and N denote a bin number and the window size respectively.

Peak extraction algorithm is applied to the obtained magnitude spectrum, which results in a list of possible candidates. The main problem of accurate frequency detection based just on the magnitude information is that the main lobe of low frequency harmonics is wider than the

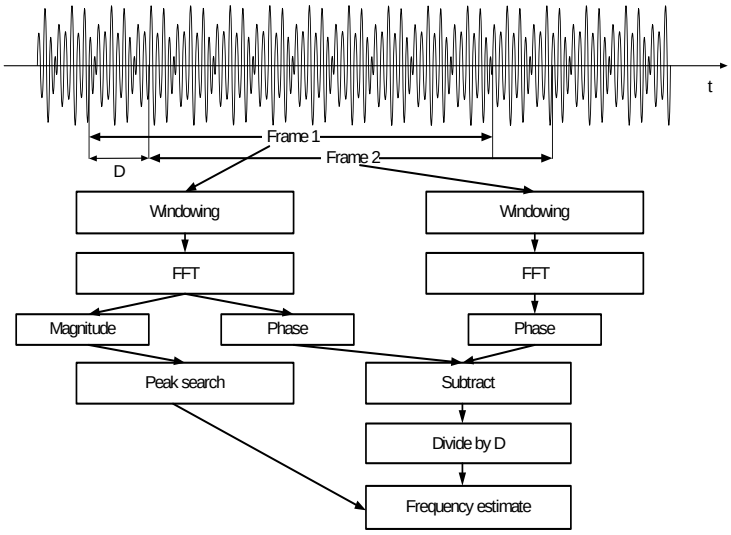


Figure 3.1: Block diagram of precise frequency estimates.

spectral resolution (and sometimes than a semitone distance). In such cases the energy of a harmonic component is distributed between adjacent bins, which represents an obstacle in the way of an accurate frequency estimation.

To cope with the above-mentioned problem, a second DFT is applied on the signal weighted by the same window, shifted by D samples, from which the difference of the two given phases divided by the time interval of D samples is calculated as follows:

$$\omega(D, N, t_0) = \frac{\arg X_w[t_0 + D, k] - \arg X_w[t_0, k]}{D} \tag{3.4}$$

The time interval D is chosen so that the phase change for the maximum frequency is less than 2π. Analyzing the obtained spectra in terms of phase-change allows for determining frequencies of harmonic components in a more accurate way, since all the adjacent bins containing the energy of a single harmonic have the same degree of phase change (see fig. 3.2).

Now, information obtained from peak-search algorithm is combined with phase-change spectrum in order to provide the final estimation. Positions of all possible candidates are checked in terms of the flatness of the corresponding frequency intervals in the phase-change spectrum.

A set of detected harmonics is compared to the table of nominal frequencies. Mean value and standard deviation of closest log-distance (based on a semitone metric) to the nearest nominal frequency are calculated in order to determine the mis-tuning and the consequent consistency of the estimate. Once this procedure has been applied, a new value is assigned to the reference frequency, which is subsequently used for feature extraction. For example, frequency of "A4" is set to 443Hz and frequencies of all the other notes are determined according to equally tempered intervals.

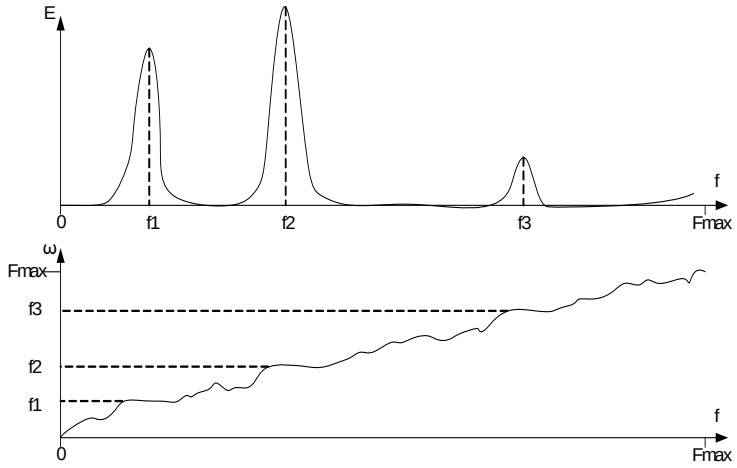


Figure 3.2: Magnitude and Phase-change spectrum.

3.3 PQMF-based chroma features

This section describes a novel approach to chroma extraction, which is based on the Pseudo-Quadrature Mirror Filters (PQMF) filter bank.

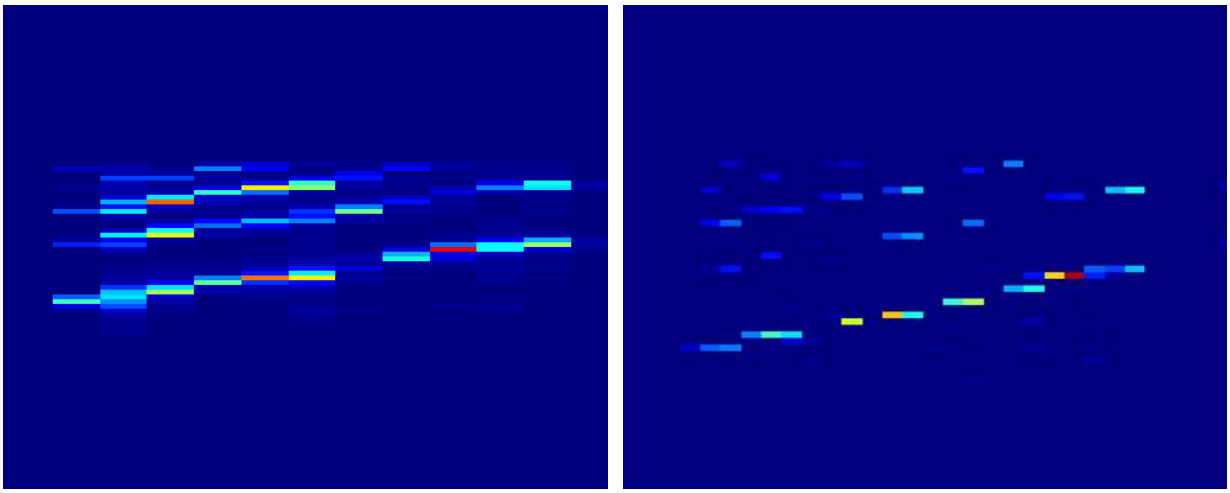
Chroma features can be extracted from audio in several different ways. The first option, which is the most common is to transform the audio to the frequency domain by means of Short-Time Fourier Transform (STFT) or Constant-Q Transform and subsequent assigning energy from different spectral bins to chroma bins [19, 21]. When performing chroma extraction using transform to frequency domain, the signal in a given analysis frame is assumed to be stationary and it is also assumed that no note transitions occur inside it. Transients and noise may cause energy assignment to some frequencies that do not occur in the signal. Due to this assumptions, the analysis frame should be short enough. At the same time, frame size should be long enough to provide reasonable frequency resolution. A trade-off between frequency resolution and stationarity should be made for a particular task. The most common frame lengths for capturing spectral content to form chroma vectors are 96ms - 360ms. As a rule, to provide smoothed feature sequence a high overlap ratio (50% – 90%) with subsequent median filtering or averaging is applied. However, using such window lengths introduces inaccuracies with rapidly changing notes. On the other hand, short window lengths do not provide reasonable frequency resolution.

An alternative way to extract chroma is to pass the analyzed signal through a multirate filter bank [33], [50]. In [50], IIR multirate filter band is proposed to derive chroma features. The filter bank is designed so that the passband of each filter is equal to a semitone width and corresponds to a certain note. Energies from different filters that correspond to the same pitch class are summed up resulting in chroma representation.

In this section, we propose a novel method that is based on multirate PQMF filter bank and subsequent periodicity estimation in the output of each filter. As opposed to the approach of Müller [33], passband of each filter is greater than semitone distance. We propose sample-by-sample periodicity estimation technique that can reflect close to instant frequency changes. Feature extraction process starts from passing the signal through a multirate filter bank. An accurate periodicity estimation is then performed on each filter output. It is assumed that features derived from this periodicity analysis reflect harmonic properties of the signal in a better way.

In the following sections, PQMF filter bank configuration is introduced and the proposed periodicity estimation technique is briefly described.

3.3.1 PQMF filter bank



(a) Pitch class profile extracted with the help of DFT of length 182 msec and 50% ovelap.

(b) Pitch class profile extracted with the help of suggested approach with the frame length analysis of 23 msec.

Figure 3.3: Comparison of DFT chroma and PQMF-based chroma features.

Quadrature Mirror Filters (QMF) is a class of perfect reconstruction filter banks that divide frequency range into 2 channels. In practical applications sometimes more channels than 2 are needed. One of the possible decisions is to build a QMF-tree or to use alternative filter banks.

A Pseudo-QMF solution, an extension of QMF, is a near perfect reconstruction filter bank that was developed and successfully used for encoding audio in MPEG layer I and II formats. It consists of N filters with equal passband bandwidths. In PQMF filter bank aliasing cancellation occurs only between adjacent bands [51].

In our approach a PQMF solution with 32 filters is adopted. Each filter has 512 taps. The impulse response of the prototype filter $h[n]$ is shown in figure 3.4. Filter coefficients $h_k[n]$ for $k-$th filter can be obtained as shown below:

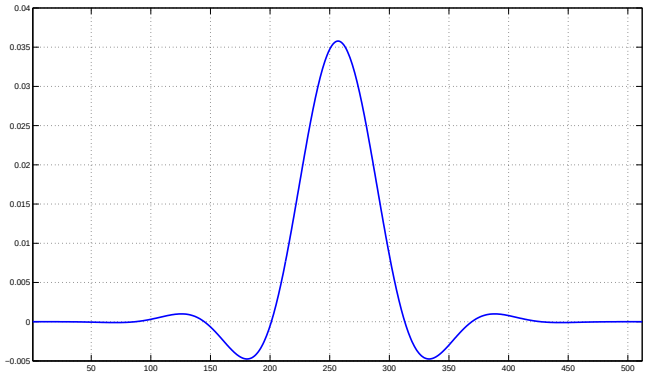


Figure 3.4: Impulse response of the PQMF prototype filter $h[n]$.

$$h_k[n] = h[n] \cos\left[\left(k + \frac{1}{2}\right)(n - 16)\frac{\pi}{32}\right] \tag{3.5}$$

Filterbank configuration

The proposed novel approach for chroma extraction with the help of filter bank analysis is based on high-precision periodicity estimates in the output of each channel. There are some conditions to be met when designing a filter bank. Passband bandwidth of the selected channels should be compared to the frequency distance between adjacent semitones. It is desirable to have filters with narrow passband bandwidth to perform better separation of the harmonics. Since the semitone distance increases exponentially with the frequency and passband bandwidth is constant in all PQMF channels, a multirate filter bank was designed. In a multirate filter bank different channels are operated at different sampling rates. Thus, starting from a prototype filter one can design a filter bank with the desirable channel passband properties.

Audio analysis starts with downsampling and filtering through a number of channels. PQMF channels, sampling frequencies and passband bandwidths are presented in Table 3.1. Magnitude responses of the first 14 filters are depicted in Figure 3.5.

The outputs of all the filters are synchronized by taking into account the delay time of each output. In the next stage, each channel output is analyzed for periodicities as described in the following section.

3.3.2 PQMF-based chroma

In this section, a new chroma vector calculation method is outlined. It is based on the analysis of the output of PQMF filter bank described in section 3.3.1. As was shown in Table 3.1, the

Sampling frequency (Hz)	Channel number	Start frequency (Hz)	End frequency (Hz)
800	4	50	62.5
800	5	62.5	75
800	6	75	87.5
800	7	87.5	100
1600	4	100	125
1600	5	125	150
1600	6	150	175
1600	7	175	200
3200	4	200	250
3200	5	250	300
3200	6	300	350
3200	7	350	400
6400	4	400	500
6400	5	500	600
6400	6	600	700
6400	7	700	800
16000	3	750	1000
16000	4	1000	1250
16000	5	1250	1500
16000	6	1500	1750
16000	7	1750	2000

Table 3.1: Filter bank configuration

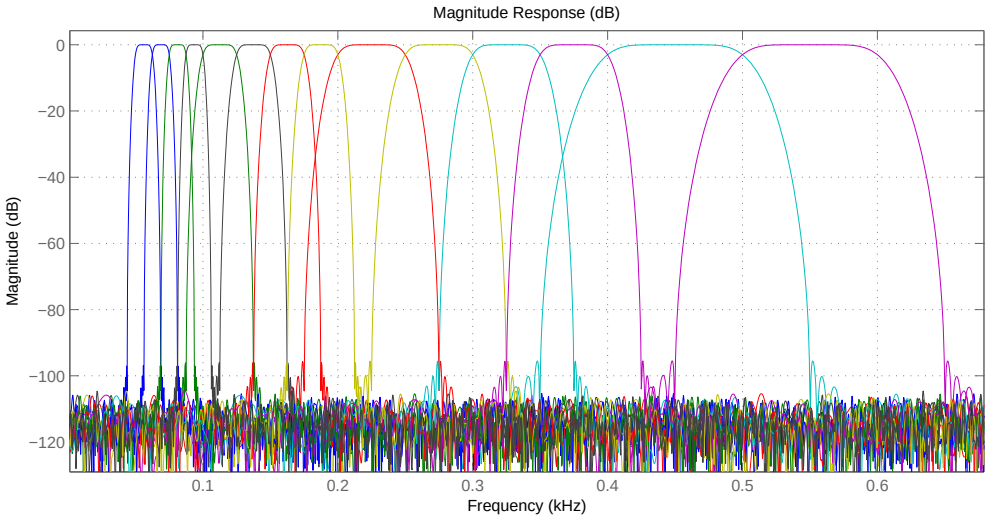


Figure 3.5: Magnitude response of the first 14 PQMF filters.

passband of output channels is greater than a semitone distance. In order to derive chroma representation, further analysis is needed. Output of each filter is analyzed for periodicities in order to estimate the frequency that corresponds to the dominant amount of energy in a given subband.

There are a lot of different approaches to periodicity estimation proposed in the literature [52]. They are based on time-domain or frequency-domain analysis. In our approach we utilize time-domain analysis that is based on accurate sample-by-sample periodicity estimation. Normalized Cross-Correlogram (NCC) is obtained and analyzed for periodicities, which proved to be effective for pitch extraction from speech signals [53].

Cross-correlogram basics

A variety of time domain methods for pitch estimation of speech signals were presented in [52]. Chung and Algazi [54] described the usage of auto-correlation and cross-correlation functions for the task. Our approach is based on the works of Medan et al. [53] and De Mori and Omologo [55]. The above mentioned works aimed at extracting pitch from speech pronounced by a single speaker. Here we adapt this methodology to multi-pitch context. This is achieved by splitting the frequency bandwidth of the signal into several subbands as described in section 3.3.1, and applying cross-correlation analysis on each channel separately.

Let $x(n)$ be a discrete signal in the time domain sampled at a sampling frequency F_s. For each time instant $t_0 = n_0 \cdot F_s$ two vectors of samples are defined as follows:

$$l_{N,n_0}(n) = x(n - N + n_0), 0 < n \leq N \tag{3.6}$$

$$r_{N,n_0}(n) = x(n + n_0), 0 < n \leq N \tag{3.7}$$

Here $l_{N,n_0}(n)$ and $l_{N,n_0}(n)$ denote left and right contexts of length N samples at the time instant n_0. Figure 3.6 shows an example of right and left contexts of different lengths N_1, N_2 and N_3.

Let us assume that in the given intervals the signal is periodic with period P. In the general case, P is a fractional number of samples that can be expressed as $P = \frac{T}{F_s}$ where T is a period in seconds. Due to the fact that we operate on the filtered signal the potential periodicity range can be determined by the frequency values that lie inside the passband interval of the given channel:

$$f_L < f < f_R \tag{3.8}$$

$$P = \frac{1}{f \cdot F_s} \tag{3.9}$$

Here f_L and f_R are the left and the right frequencies that define passband bandwidth of the filter.

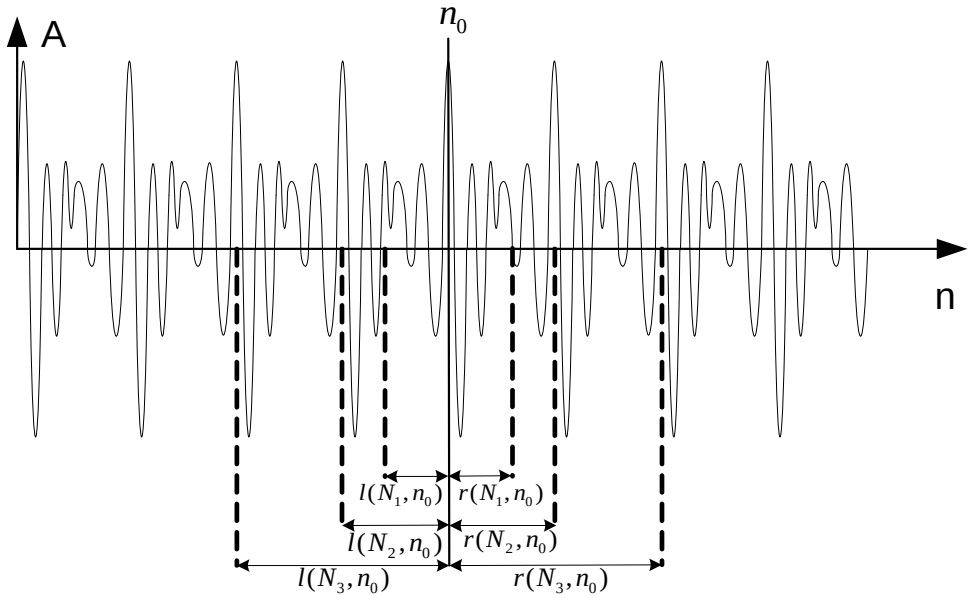


Figure 3.6: Example of left and right contexts of different lengths at the time instant n_0.

The normalized cross-correlation coefficient between the left and the right contexts is computed as follows:

$$S\left(N, n_0\right) = \frac{l_{N,n_0}(n) \cdot r_{N,n_0}(n)}{|l_{N,n_0}(n)| \cdot |r_{N,n_0}(n)|} \tag{3.10}$$

Instantaneous period P' at the time instant n_0 can be estimated as:

$$P' = \underset{N}{argmax}\{S(N, n_0)\} \tag{3.11}$$

High values of normalized cross-correlation can be observed in the multiples of the period. In figure 3.6 one can see that the context lengths of N_2 and N_3 samples provide high cross-correlation coefficient between the left and the right contexts, while using the context lengths of N_1 samples results in lower cross-correlation value. Due to the fact that we have limited range of possible period values defined by Equations (3.8) and (3.9), the ambiguity in the multiples of the period is avoided.

Figure 3.7 shows a cross-correlogram visual representation of one of the filter bank channels output. The first part of the cross-correlogram (0s – 3s) regards a strongly periodical signal with the period of 23.4 samples. In the second part (3s – 6s) the period is 18.9 samples. While in the interval from 6s to the end of the excerpt detected periodicity has evident peak in the 5-th or 6-th multiple of the period.

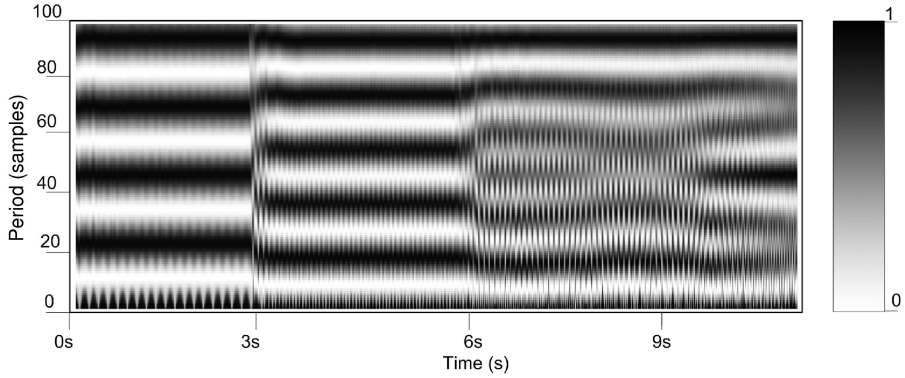


Figure 3.7: Crosscorrelogram for one of the filterbank channels.

3.3.3 From cross-correlogram to chroma representation

For a given frame, once estimated the periodicity value P_j for each sample j, the obtained data are used to derive chroma vector. For a window size d and for each frame i, the frequency f_i and the RMS energy E_{rmsi} are computed as shown below:

$$f_i = \frac{\sum_{j=j'}^{j''} \frac{F_s}{P_j}}{j'' - j' + 1} \tag{3.12}$$

$$E_{rmsi} = \frac{\sum_{j=j'}^{j''} x(j)^2}{j'' - j' + 1} \tag{3.13}$$

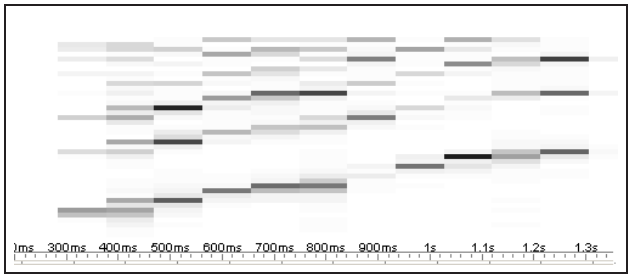
where $j' = \lceil i \cdot d \cdot F_s \rceil$ and $j'' = \lfloor (i+1) \cdot d \cdot F_s \rfloor$. The E_{rmsi} portion of energy is added to the chroma bin $c(f_i)$ that corresponds to the detected frequency f_i based on the following equation:

$$c(i) = 12\log_2\left(\frac{f_i}{f_{ref}}\right) + 69 \tag{3.14}$$

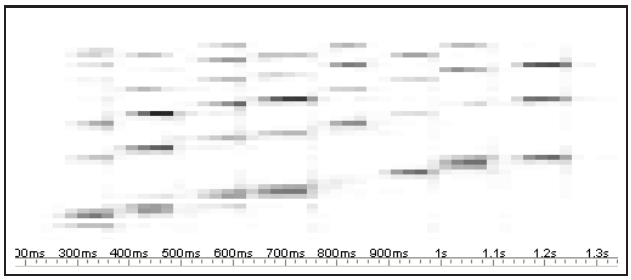
where f_{ref} is the reference frequency of the A4 note.

This operation is applied to all the filter bank outputs, and as a result a chroma representation is obtained, where a 12-bin chroma vector corresponds to each frame i.

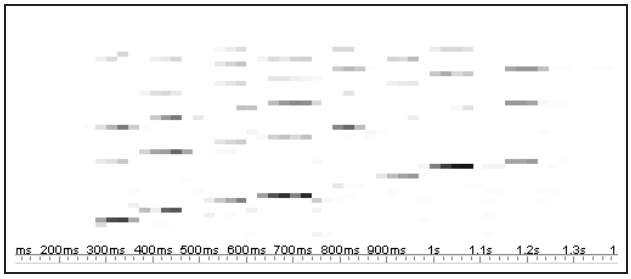
Figure 3.8a shows the example of standard chroma using DFT computed on a window length 182ms with 50% overlap. The given window length in some cases does not allow for precise capturing the harmonic properties, since inside such a long window analysis some note transitions are likely to occur. This leads to the distribution of spectral energy among adjacent chroma bins.



(a) Standard chroma of length 182 msec and 50% ovelap



(b) Standard chroma of length 46 msec and 50% ovelap



(c) PQMF-based chroma of length 23 msec and 0% ovelap

Figure 3.8: Unwrapped chroma vectors extracted from a short note passage by means of different approaches.

Figure 3.8b depicts chroma for the same same signal, but extracted with the help of DFT of 46 msec with 50% overlap. In this case the analysis window size provides the necessary time-domain resolution for capturing rapid note changes, but on the other hand, low spectral resolution causes wide lobes of the spectral components that leads to spectral leakage.

The proposed approach to chroma feature vectors extraction is based on PQMF filtering and subsequent periodicity detection, and in general does not introduce the above-mentioned drawbacks. Chroma vectors extracted with the new technique are displayed in Figure 3.8c.

3.4 Time-frequency reassigned chroma features

In the past few years a lot of different techniques for accurate and relevant feature extraction in automatic chord recognition have been proposed. In this section we propose another method for chroma extraction that is based on the Time-Frequency Reassigned spectrum.

Feature extraction process is aimed at transforming a given waveform into a representation that captures desirable properties of an analyzed signal. A lot of acoustic features is derived from some kind of time-frequency representations, which can be obtained by mapping audio signal from one-dimensional time domain into two-dimensional domain of time and frequency. Spectrogram is one of the most widely spread time-frequency representations that has been successfully used in a variety of applications, where spectral energy distribution changes over time.

Time-frequency reassignment technique was initially proposed by Kodera et al. [56]. The main idea behind TFR technique is to remap spectral energy of each spectrogram cell into another cell that is the closest to the true region of support of the analyzed signal. As a result, "blurred" spectral representation becomes "sharper", that allows one to derive spectral features from reassigned spectrogram with much higher time and frequency resolution. Some papers have already investigated the usage of reassigned spectrogram in different tasks, such as sinusoidal synthesis [57], cover song identification [58] and many others.

Now some mathematical foundations for the TFR technique are provided. Let $x(n)$ be a discrete signal in the time domain sampled at a sampling frequency F_s.

At a given time instant t, STFT is performed on the signal weighted by a window function $w(n)$ as in the following

$$X(t,k) = \sum_{n=0}^{M-1} w(n)x(n+t)e^{-2\pi jnk/M}, \tag{3.15}$$

where k and M denote a bin number and the window size respectively. Spectrogram is derived from (3.3) as shown in (3.16).

$$P(t,k) = |X(t,k)|^{\delta} \tag{3.16}$$

where δ is equal to 2. The majority of chromagram extraction techniques uses this representation for mapping spectral energies to chroma bins, ignoring phase information as in the following

$$n(k) = 12\log_2\left(\frac{f_k}{f_{ref}}\right) + 69, n \in \Re^+, \tag{3.17}$$

where f_{ref} denotes the reference frequency of "A4" tone, while f_k and n are the frequencies of the Fourier transform and the semitone bin scale index, respectively.

On the other hand, the result of STFT $X(t,k)$ can be presented in the following form:

$$X(t,k) = M(t,k)e^{j\phi(t,k)}, \tag{3.18}$$

where $M(t,k)$ is the magnitude, and $\phi(t,k)$ the spectral phase of $X(t,k)$. As was shown in [59], reassigned time-frequency coordinates $(\hat{t},\hat{\omega})$ can be calculated as

$$\hat{t}(t,\omega) = -\frac{\partial\phi(t,\omega)}{\partial\omega} \tag{3.19}$$

$$\hat{\omega}(t,\omega) = \omega + \frac{\partial\phi(t,\omega)}{\partial t} \tag{3.20}$$

Efficient computation of $\hat{t}(t,\omega)$ and $\hat{\omega}(t,\omega)$ in the discrete-time domain was proposed by Auger and Flandrin [60] and takes the following form:

$$\hat{t}(t,\omega) = t - \Re\left\{\frac{X_{\mathcal{T}w}(t,\omega)\cdot X^*(t,\omega)}{|X(t,\omega)|^2}\right\} \tag{3.21}$$

$$\hat{\omega}(t,\omega) = \omega + \Im\left\{\frac{X_{\mathcal{D}w}(t,\omega)\cdot X^*(t,\omega)}{|X(t,\omega)|^2}\right\} \tag{3.22}$$

where $X_{\mathcal{D}w}$ is the STFT of the signal weighted by a frequency-weighed window function, $X_{\mathcal{T}w}$ is the STFT of the signal weighted by a time-weighed window function ([59]). Reallocating spectral energy from spectrogram coordinate (t,w) to $(\hat{t},\hat{\omega})$ concludes the reassignment operation. As a result more precise estimates of spectral energy distribution are obtained. However, reassigned spectrogram can be noisy. A random energy can be located in points where there are no obvious harmonic or impulsive components. The principle of the reassignment technique is to reallocate energy from the geometrical center of the analysis window to the "center of gravity" of the spectral component this energy belongs to. Meanwhile, in some spectral regions, where there are no dominant components, large energy reassignment both in time and frequency can be observed. In order to obtain a better spectral representation and to refine the spectrogram keeping the energy of harmonic components and deemphasizing that of noisy and impulsive components, the following condition should be met [61]

$$\left|\frac{\partial^2\phi\,(t,\omega)}{\partial t\partial\omega} + 1\right| < A \tag{3.23}$$

where A is the tolerance factor, which defines the maximum deviation of the acceptable spectral component from a pure sinusoid. The optimal value of A depends on a particular task and can be empirically determined. Fullop and Fitz reported in [62] that 0.2 is often a reasonable threshold for speech signals.

As for the impulsive part of the spectrogram, the filtering condition takes the following form:

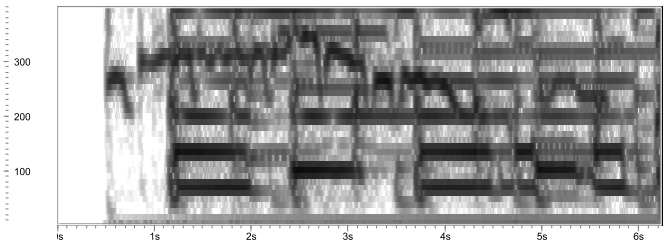
$$\left|\frac{\partial^2\phi\left(t,\omega\right)}{\partial t\partial\omega}\right| < A \tag{3.24}$$

Efficient computation of $\frac{\partial\phi^2(t,\omega)}{\partial t\partial\omega}$ is given in [59] and can be expressed as follows

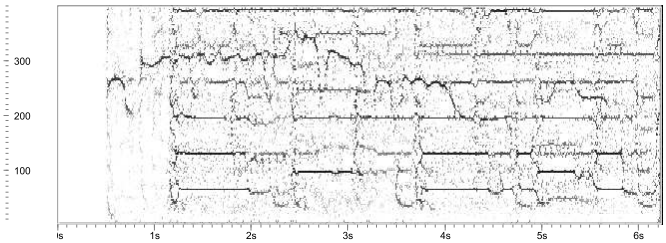
$$\frac{\partial^2\phi\left(t,\omega\right)}{\partial t\partial\omega} = \Re\left\{\frac{X_{\mathcal{TD}w}(t,\omega)X^*(t,\omega)}{\left|X(t,\omega)\right|^2}\right\} - \Re\left\{\frac{X_{\mathcal{T}w}(t,\omega)X_{\mathcal{D}w}(t,\omega)}{X^2(t,\omega)}\right\} \tag{3.25}$$

where $X_{\mathcal{TD}w}(t,\omega)$ is the STFT of the signal weighted by time-frequency-weighed window function ([59]).

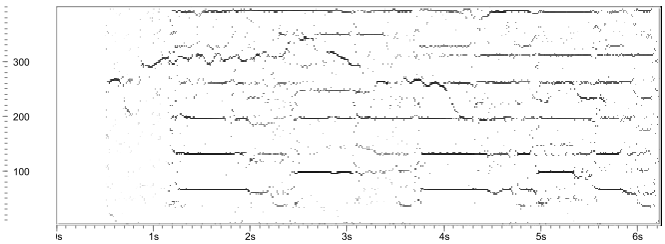
Comparison of spectrogram, reassigned spectrogram and "refined" reassigned spectrogram for an excerpt from "Girl", the Beatles is provided in Figure 3.9. All spectrograms are computed using Hanning window of 192 ms with 90% overlapping.



(a) Spectrogram



(b) Reassigned spectrogram



(c) Harmonic reassigned spectrogram with tolerance factor set to 0.4

Figure 3.9: Time-Frequency representation of an excerpt from "Girl", the Beatles. All spectrograms are computed using Hanning window of 192 ms with 90% overlapping.

3.5 Towards reducing dimensionality

In this section we try to explore the advantage of the dimensionality reduction on the feature space. We consider two different approaches to dimensionality reduction. The first approach is based on the computation of Tonal Centroid (TC), a 6-dimensional feature that proved to be quite effective for the problem of dividing audio into harmonically homogeneous segments [63, 64]. Some attempts were made to use TC for chord recognition [20]. The second approach is based on the idea of transforming feature vectors of chroma family by Inverse Discrete Cosine Transform (IDCT).

3.5.1 Tonal centroid

Tonal centroid was first introduced by Harte et al. in [63], who proposed to use it to detect harmonic changes in audio.

In chord recognition domain the usage of TC was investigated by Lee et al. [20]. They showed that using TC as feature set instead of conventional chroma leads to a significant increase in recognition rate. The experiments were carried out on the first two Beatles albums as well as on a short set of classical excerpts by Bach and Haydn. Another application of TC for chord recognition was suggested by Harte et al. in [64], where the algorithm of detecting harmonic changes introduced in [63] was utilized as a pre-processing step to determine chord boundaries. Obtained segmentation information is used in the next step to obtain average chroma vector for each segment and perform classification by template matching.

Conceptually, TC is based on Tonnetz, a harmonic network, where notes with closer harmonic relations have smaller distance. Tonnetz plane is infinite. However, some music-related tasks, e. g. chord recognition, assume enharmonic and octave equivalence. The computation of TC is based on the transformation on chroma vector into 6-D space, where three pairs of coordinates assume projection onto three different circles: major thirds, minor thirds and fifths [63]. The computation is performed by multiplying of chroma vector c_n and transformation matrix Φ as follows:

$$\varsigma_n(d) = \frac{1}{\|c_n\|_1} \sum_{l=0}^{11} \Phi(d,l) c_n(l) \qquad 0 \leq d \leq 5, 0 \leq l \leq 11 \tag{3.26}$$

Here $\|c_n\|_1$ is the L_1 norm of c_n and matrix Φ is

$$\Phi(d,l) = \begin{bmatrix} \Phi(0,l) \\ \Phi(1,l) \\ \Phi(2,l) \\ \Phi(3,l) \\ \Phi(4,l) \\ \Phi(5,l) \end{bmatrix} = \begin{bmatrix} r_1 \sin l\frac{7\pi}{6} \\ r_1 \cos l\frac{7\pi}{6} \\ r_2 \sin l\frac{3\pi}{2} \\ r_2 \cos l\frac{3\pi}{2} \\ r_3 \sin l\frac{2\pi}{3} \\ r_3 \cos l\frac{2\pi}{3} \end{bmatrix} \qquad 0 \leq l \leq 11 \tag{3.27}$$

3.5.2 Application of Inverse Discrete Cosine Transform

Another technique for reducing dimensionality we investigated is the Inverse Discrete Cosine Transform (IDCT). It proved to be quite effective in speech processing domain [65]. IDCT coefficients $\psi(n)$ are obtained as

$$\psi(n) = \sum_{k=0}^{N-1} x(k) \cos\left(\frac{\pi k\,(2n+1)}{2N}\right) \qquad 1 \leq n \leq N_c \tag{3.28}$$

where $x(k)$ is the input vector, N is the number of bins in $x(k)$ and N_c is the number of output IDCT coefficients.

We set up experiments with $N_c = 16$ and we use combined chroma vector c_{com} as the input vector for IDCT $x(k)$. c_{com} is comprised of bass c_b and treble c_t chroma vectors and has 24 dimensions. In order to investigate influence of the chroma components order inside c_{com} we build two different input vectors c_{com1} and c_{com2}.

$$c_{com1}(k) = \begin{bmatrix} c_b(0) & c_b(1) & \ldots & c_b(11) & c_t(0) & c_t(1) & \ldots & c_t(11) \end{bmatrix} \tag{3.29}$$

$$c_{com1}(k) = \begin{bmatrix} c_b(0) & c_t(0) & c_b(1) & c_t(1) & \ldots & c_b(11) & c_t(11) \end{bmatrix} \tag{3.30}$$

We also investigate system performance using mean subtraction technique, that proved to provide more robust features in speech processing [66]. Mean subtraction is a post-processing step, which includes the following actions. At first, mean value of feature vectors extracted from the whole piece of audio is estimated . Then, the obtained mean value is subtracted from each feature vector. In the experimental section we will investigate the efficiency and usefulness of mean vector subtraction for IDCT features in the chord recognition task.

Chapter 4

System architecture

This chapter is concerned with the proposed statistical methods to automatic extraction of chords from audio. The structure of the chapter is as follows: Section 4.1 introduces acoustic modeling approach adopted here. Section 4.2 describes language modeling techniques. Application of standard and factored language models is outlined. Finally, general overview of the proposed chord recognition system is given.

4.1 Acoustic modeling using multi-stream HMMs

This section refers to the acoustic modeling mechanisms applied here for chord recognition. Acoustic modeling part is based on HMMs and is quite similar to the one described in [67] and [25]. However, these approaches are extended to the usage of a more general version of HMMs with multi-stream observation layer. A similar technique was used in [27], where a dynamic Bayesian network was configured to contain bass and treble observable layers.

As in the case of a single-stream HMM, a multi-stream HMM consists of a number of states N, Each state j characterized by its observation probability distribution $b_j(o_t)$ that defines the probability to emit observation symbol o_t at time instant t. An important parameter is the transition matrix a_{ij} that determines the probability of transition from state i to state j. Continuous density models are used in which each observation probability distribution is represented by a mixture of multivariate Gaussians. In the multi-stream HMM, the related observation layer consists of multiple streams and $b_j(o_t)$ can be expressed as

$$b_j(o_t) = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_{js}} c_{jsm} \mathcal{N}\left(o_{st}; \mu_{jsm}, \Sigma_{jsm}\right) \right]^{\gamma_s}, \tag{4.1}$$

where M_{js} denotes the number of mixture components in state j for stream s, c_{jsm} is the weight of the m-th component and $\mathcal{N}\left(o_{st}; \mu_{jsm}, \Sigma_{jsm}\right)$ is a multivariate Gaussian with mean

vector μ and covariance matrix Σ. Each Gaussian component $\mathcal{N}\left(o_{st}; \mu_{jsm}, \sum_{jsm}\right)$ can be expressed as

$$\mathcal{N}(o; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left(-\frac{1}{2}(o-\mu)'\Sigma^{-1}(o-\mu)\right) \tag{4.2}$$

where n is the dimensionality of observation o. The term γ_s is a stream weight. Varying this parameter allows one to emphasize or deemphasize the contribution of a particular stream.

Figure 4.1 depicts a typical structure of multi-stream HMM with three hidden emitting states and S observation streams.

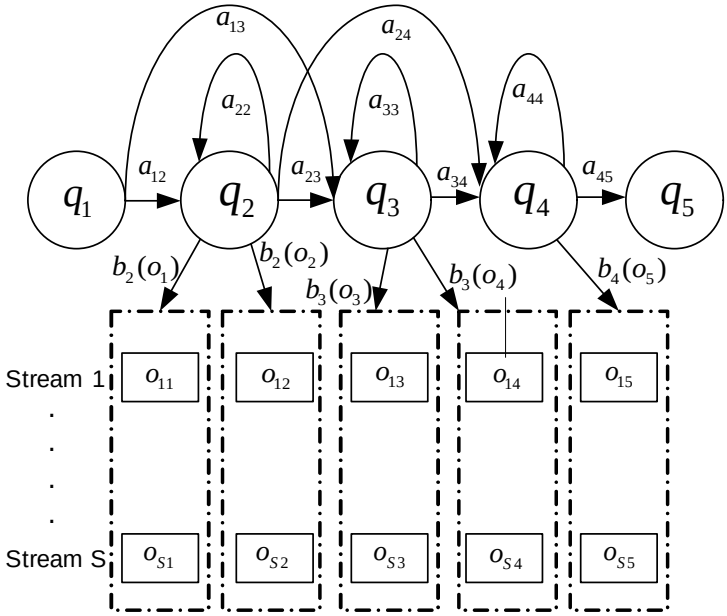


Figure 4.1: Structure of multi-stream HMM with three hidden emitting states

Training is performed for each chord type from the predefined dictionary, resulting in a separate left-to-right HMM. A chord type represents chords with a given set of intervals between constituent notes regardless of the root note, e.g. major, minor. Each model consists of 1 – 3 emitting hidden states. Observation probability distributions are learned from data in the training stage. Feature vector components are assumed to be uncorrelated with one another, so the covariance matrix has a diagonal form.

Trained multi-stream HMMs are then connected as shown in figure 4.2. An insertion penalty is introduced to influence the transition probability between chords. Varying the insertion penalty allows for obtaining labels with different degrees of fragmentation, as typically done in speech recognition tasks. As was shown in [25], the insertion penalty (or self-transition probability in [68]) can have a significant impact on the overall performance.

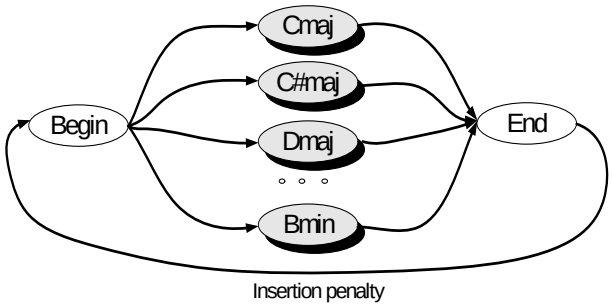


Figure 4.2: Connection scheme of trained models for decoding.

In the experimental part two different HMM configurations are evaluated – baseline and multi-stream one. The former configuration includes one observation stream, where emitted symbols are chroma vectors. In the latter case, an additional observation bass chroma stream is added.

4.2 Language Modeling

A lot of different statistical language models have been proposed over years. The most successful among them appeared to be finite state transducers. In Natural Language Processing N-grams are used for word prediction. Given $N-1$ predecessors, they can provide the probability of N-th element appearing. Language models have a variety of applications such as automatic speech recognition and statistical machine translation. The main goal of language modeling can be explained as follows: having a sentence, which consists of K words ($w_1, w_2, ...w_K$), generate a probability model $p(w_1, w_2, ...w_K)$. In most common cases it can be expressed as

$$p(w_1, w_2...w_K) = \prod_t p(w_t|w_1, w_2...w_{t-1}) = \prod_t p(w_t|h_t) \tag{4.3}$$

where h_t is the history sufficient for determining the probability of w_t word. In standard N-gram models the history consists of the immediately adjacent $N-1$ words. For example, in 3-gram model the probability of current word can be expressed as: $p(w_t|w_{t-1}, w_{t-2})$.

While estimating language model parameters, there exists the problem of sparse data. It is caused by the impossibility of producing maximum likelihood estimate of the model, because all combinations of N-word sequences are unlikely to be found in the training corpus. Since any training corpus is limited, some acceptable sequences can be missing from it, which leads to setting zero probability to plenty of N-grams. In order to cope with the problem, different techniques, such as back-off, smoothing and interpolation are used [69–71]. The main principle of back-off is to rely on lower-order model (e.g $p(w_t|w_{t-1})$) if there is zero evidence for higher-order (e.g. $p(w_t|w_{t-1}, w_{t-2})$) model. The order of dropping variables is known as back-off

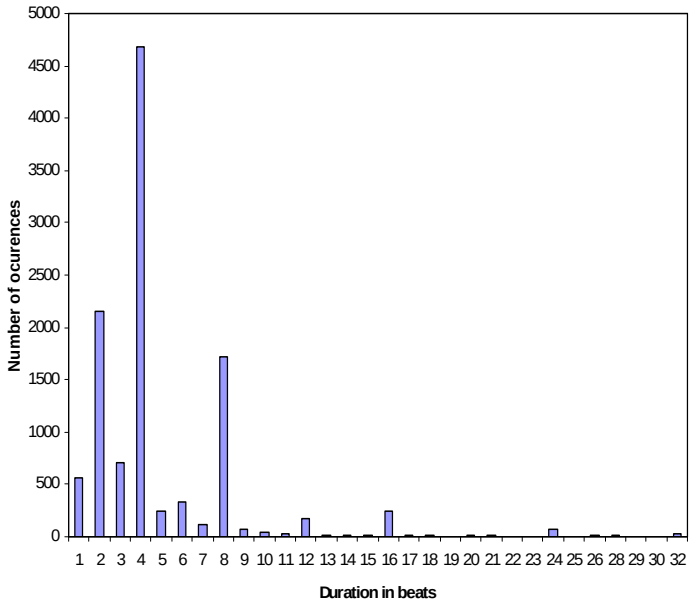


Figure 4.3: Chord Duration Histogram.

order. In the case of standard language models it is obvious that information taken from older predecessor will be less beneficial and it should be dropped prior to other predecessors.

In the proposed approach we draw a direct analogy between a sentence in speech and a part of a song. The above-described strategy can be successfully used in chord sequences modeling. In this case a chord is the equivalent of a word and the sequence of chords can be modeled by means of the same technique.

4.2.1 Factored language models

Western music is known to be highly structural in terms of rhythm and harmony. In order to take advantage of mutual dependency between these two phenomena, we have studied the interrelationship between beat structure and chord durations. The number of occurrences as a function of chord duration in beats histogram is shown in figure 4.3. It is clearly seen that a large part of chord durations is correlated to the metrical structure (2, 4, 8, 12, 16, 24, 32 beats), which suggests that including also chord durations in the language model is more convenient than analyzing just a sequence of chord symbols. This can be easily done with the help of factored language models (FLMs), which treat a word (chord) as a set of factors. FLMs have been recently proposed by Bilmes and Kirchoff [72] and showed promising results in modeling highly inflected languages, such as Arabic [73].

In a Factored Language Model, a word (chord) can be represented as a bundle of factors:

$w_t = \{f_t^1, f_t^2, ..., f_t^K\}$. The probability for FLM is given in (4.4), where $\pi(f_t^k)$ is a set of variables (parents), which influence the probability of f_t^k. In our case to model chord sequences we use two factors: chord label C_t and chord duration D_t: w_t = $\{C_t, D_t\}$.

$$p(w_t|h_t) = \prod_k p(f_t^k|\pi(f_t^k)) \tag{4.4}$$

As opposed to standard language models, where older predecessors give less relevant information at the given time instant, in FLMs there is no obvious order to drop parents $\pi(f_t^k)$. There are a lot of possibilities to choose less informative factors to drop among the others. Moreover, keeping some factors of older predecessors can be of greater benefit than keeping the value of some other factors, which are more relevant to the given time instant. One of the possible solutions is to use "generalized parallel back-off", which was initially proposed and well described by Bilmes and Kirchoff [72]. The main idea is to back-off factors simultaneously. The given set of back-off paths is determined dynamically based on the current values of the variables. (For a more detailed description, see [72]).

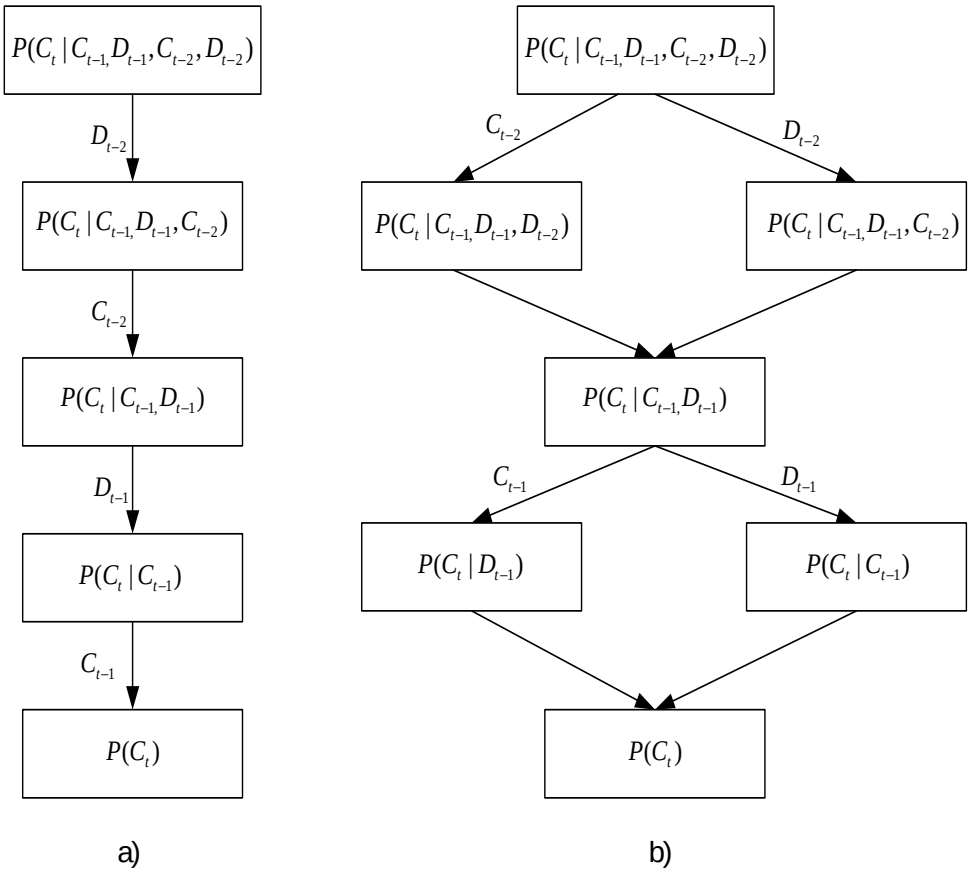


Figure 4.4: Standard back-off (a) and parallel back-off (b) graphs for tri-gram LM.

At the experimental stage we explore the standard back-off (a) and the parallel back-off (b) techniques, whose graphs are presented in figure 4.4. In both cases the chronological order is kept, while in the standard back-off case a higher priority to the factor of chord symbol is assigned. The arrows are marked with the factor being dropped at the current back-off step; blocks include the variables that influence the probability of chord label being estimated.

4.3 System overview

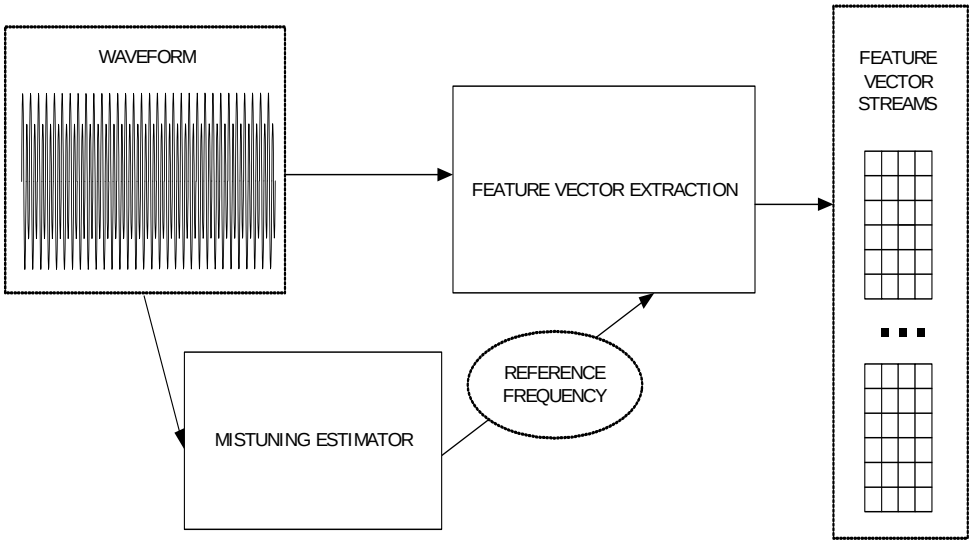


Figure 4.5: Feature extraction block diagram

This section is concerned with an overview of the proposed chord recognition system. Figures 4.5 and 4.7 show the two main blocks of the chord recognition system. Feature extraction, including mistuning estimation, produces feature vector streams that are subsequently processed by decoder. A fundamental step regards model training based on the application of Baum-Welch algorithm as depicted in Figure 4.6.

In the proposed chord recognition system chroma features are used to model emission probabilities, while HMMs are used to model chord progressions. Three main blocks can be emphasized, feature extraction (Figure 4.5), training (Figure 4.6), and testing (Figure 4.7).

4.3.1 Mistuning estimation

In the general case, chroma feature is obtained by summing all the spectral energies corresponding to a given semitone bin. Central frequencies for a specific bin are calculated using the information on the reference frequency f_{ref} of "A4" note and the mapping itself is performed as shown in the Equation (3.1). The problem of the reference frequency estimation arises in

several cases. Sometimes all the instruments that participate in the music performance can be well tuned to each other and a listener would not notice any audible artifacts. However, the reference frequency f_{ref} can slightly deviate from the conventional 440Hz, which will cause incorrect mapping of the spectral energy into semitone bins and, as a result, it will lead to less accurate acoustic features. Another possible scenario could happen during the process of converting vinyl LP records and tapes from analog to digital. A slight deviation of the moving mechanism speed leads to shifting all the frequencies to a certain degree.

The importance of the f_{ref} estimation is evident [67, 74] and is an essential part of the feature extraction step. A detailed description of the detuning estimation approach is given in Section 3.2. The obtained value f_{ref} is subsequently used for the creation of the semitone bin frequency ranges.

4.3.2 Model training

In the training stage, features extracted from waveforms are first segmented according to the ground-truth labels so that each segment contains one chord. The circular permutation procedure is then applied in order to discard root information. At this point, a number of feature vector segments is collected for each chord type that are subsequently used to train HMMs. Finally, in order to obtain model parameters for all possible chords for a given chord type, another circular permutation on the mean vectors and covariance matrix of multivariate Gaussians is performed.

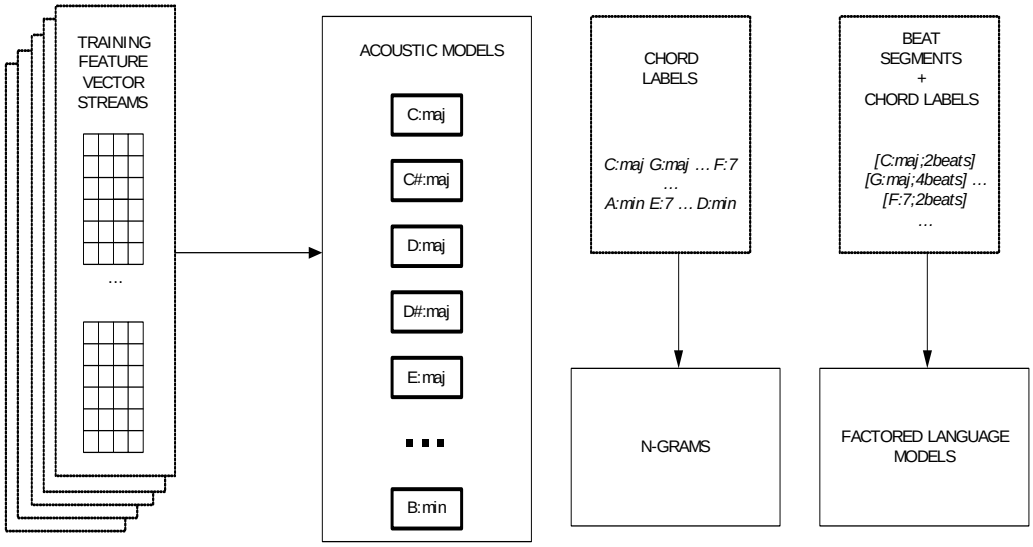


Figure 4.6: Training phase block diagram. Baum-Welch algorithm for HMM training and n-gram model parameter estimation using ground-truth labels.

In order to prevent the lack of training data (some chord types can appear only few times

in the training corpus) only two models are trained: C-major and C-minor. For this purpose, all chroma vectors obtained from labeled segments are mapped to the C-root using circular permutation. After that, mean vectors and covariance matrices are estimated for the two models. All the other models can be obtained by a circular permutation procedure.

At the same time, chord labels from training corpus are used as an input for language model parameter estimation. Language model training includes training either standard LMs or FLMs. For training standard LMs chord sequences taken from the training labels are used as input. For building text for FLM the information combined from beat extraction module and the training labels is used. Beat extraction algorithm used here was introduced by Dixon [75] and is exploited as a separate module, called *BeatRoot*[1]. For each chord symbol from ground-truth labels we estimate the duration in beats and produce an output in the form: "C-(chord type):D-(duration)". To minimize the problem of sparse data, all duration values are quantized by a relatively-small set of or integer values. Our codebook consists of the following values: 1, 2, 3, 4, 6, 8, 12, 16, 24 and 32 beats. The suggested codebook is supposed to be well-suited for the pop songs. This assumption is made on the basis of metrical analysis of the Beatles data (see fig. 4.3). The suggested scheme however might not be sufficient while modeling jazz or other genres.

In order to make our system key invariant, a key transformation technique is proposed here. In fact, the training corpus might not contain some type of chords and chord transitions due to the fact that keys with a lot of accidentals are much less widespread (G#:maj, Ab:min). Moreover, while estimating chord transition probabilities the relative change in the context of the given key (e.g. tonic – dominant – subdominant) is more relevant than exact chord names. For training data we have ground-truth table of keys for each song, while for test data we estimate key in the key detection module. Then, similar to training HMMs, by applying circular permutation, features and labels are converted to the Cmaj (in case of major key) or to Amin (in case of minor key). After the decoding procedure in order to produce final labels (in the original key of the analyzed song) obtained labels are converted back using the same scheme.

4.3.3 Decoding step

General block-scheme of decoding process is depicted in Figure 4.7.

The system can output labels in two different ways. The first option is to directly use the output of the Viterbi decoder, which is the optimal path through the hidden states of the HMMs. However, this system configuration does not use statistical modeling of chord sequences. All the chords have the same probability to be generated. We refer to this system as "No-LM" configuration. Dashed arrow in Figure 4.7 shows the process of direct deriving of chord labels

[1] http://www.elec.qmul.ac.uk/people/simond/beatroot/index.html

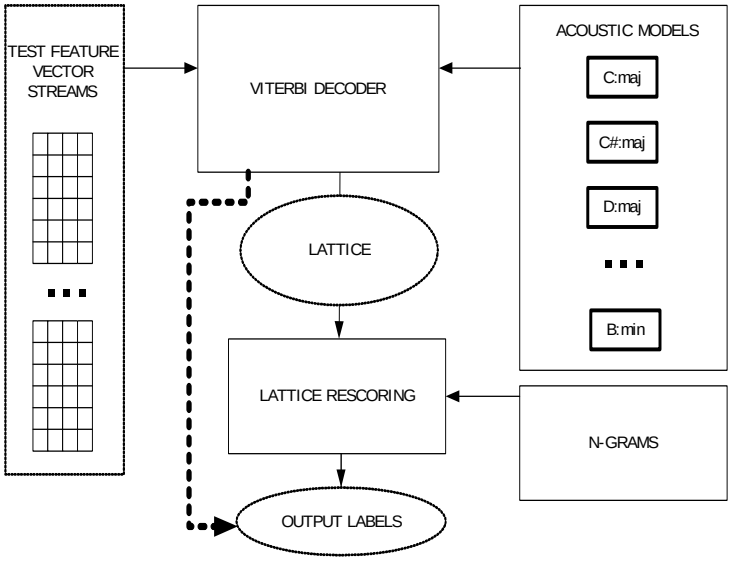


Figure 4.7: Test phase block diagram.

after Viterbi decoding.

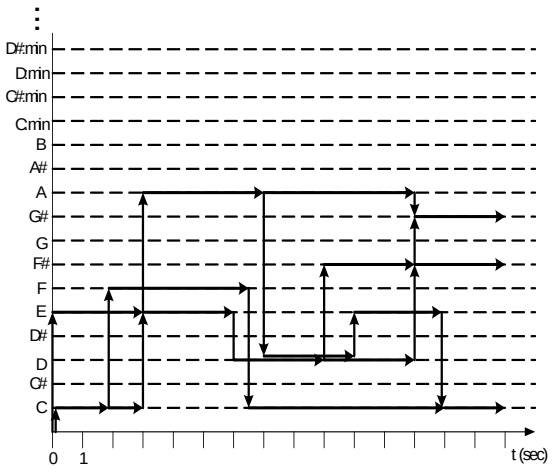


Figure 4.8: An example of a lattice.

The second system configuration involves statistical modeling of chord progressions. It will be referred to as "LM" configuration. Similar to the approach of multiple-pass decoding, which has been successfully used in speech recognition [71], the decoding procedure consists of two steps. During the first step, bigram language model is applied in the stage of Viterbi decoding, producing a lattice. A lattice can be represented by a directed graph, where nodes denote time instants and arcs are different hypotheses. Since lattices contain the information on the time boundaries, it is possible to make an estimation of duration in beats for each hypothesis. During the second step the obtained lattice is rescored applying more sophisticated language models

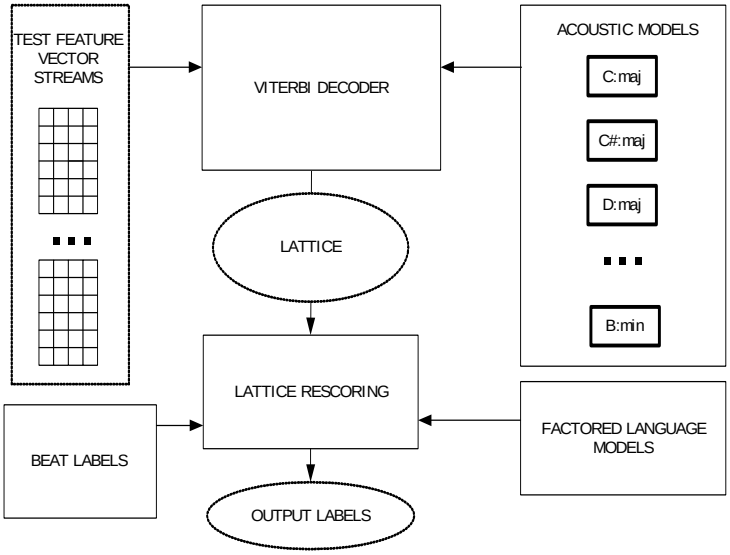


Figure 4.9: Test phase block diagram using FLMs.

(trigram and higher) on the reduced search space.

System configuration with FLMs, which will be referred to as "FLM" configuration, requires the same beat extraction procedure as in the model training step. The modified version of the decoder that makes use of FLMs is shown in Figure 4.9. The decoding scheme is also based on Viterbi decoding and subsequent lattice rescoring. Nodes in a lattice contain the time information on possible chord boundaries. Beat information is used to assign the duration factor for each chord hypothesis. The "LM" system configuration does not take into account duration factor at all. The advantage of FLM is that when applying the language model weight in the stage of lattice rescoring, chord durations contribute to the probabilities of different hypotheses in the lattice.

Standard LMs are manipulated using HTK[2] tools, while FLMs are managed using SRILM [76] toolkit, since HTK does not support this type of language models.

[2]http://htk.eng.cam.ac.uk/

Chapter 5

Experimental results

This chapter is concerned with the evaluation of chord recognition system introduced in chapter 4. At first, datasets and evaluation metrics are presented. Impact of different configuration parameters on the chord recognition performance is investigated. Then, we explore the use of Factored Language Models and compare them to the standard N-grams. Another portion of experimental results regards different feature vector solutions. We carefully examine and compare the performance of standard chroma and PQMF-based chroma introduced in Section 5.4. Then, we perform comprehensive evaluation of TRF-based chroma features that are introduced in Section 3.4. Finally, experiments with with multi-stream HMM configuration described in Section 4.1 are carried out.

5.1 Description of the dataset and evaluation metrics

5.1.1 Evaluation metrics

Chord recognition system evaluation

For evaluation, the recognition rate measure was used, which in the given case corresponds to the total duration of correctly classified chords divided by the total duration of chords, as reported in the following

$$RR = \frac{|recognized\_chords| \cap |ground - truth\_chords|}{|ground - truth\_chords|} \tag{5.1}$$

Evaluation was performed on a frame-by-frame basis, as it was done under the MIREX competition. The system can distinguish 24 different chord types (major and minor for each of 12 roots). 7th, min7, maj7, minmaj7, min6, maj6, 9, maj9, min9 chords are merged to their root triads; suspended augmented and diminished chords are discarded from the evaluation task.The

percentage of duration of discarded chords results to be 2.71% of the whole material. The proposed chord dictionary will be referred to as "maj-min" configuration.

In order to determine, whether the results produced by different systems are significantly different from each other, a Tukey's HSD (Honestly Significant Difference) test is performed. HSD test is an easy and frequently used pairwise comparison technique. HSD test finds what means are significantly different from one another. Detailed description of HSD test is provided in [77].

Chroma quality evaluation

One of the main goals of this work is to compare the effectiveness of the proposed acoustic feature set and compare the ability to carry relevant information for chord discrimination. The most obvious way to evaluate it is the chord *recognition rate (RR)* as given in the previous section. However, here we propose two additional estimates to evaluate the quality of a chroma vector – *ratio (R)* and *cosine measure (CM)*, they are are computed as proposed in [78].

Let $c(n)$ be an unwrapped chroma vector extracted from a chord sample that was generated from a set of notes e. The *R* estimate is the ratio of the power in the expected semitone bins, over the total power of that frame. The expected semitone bins include the bins of the fundamentals and 3 partials for every note from set e.

To estimate *CM* a chroma template $y(n)$ is built so that its values are set to 1 in the chroma bins that correspond to the fundamentals and to 0.33 in the chroma bins that correspond to the first 3 partials. The CM estimate is then computed as $CM = \frac{\langle y \cdot c \rangle}{\|y\| \|c\|}$, where $\langle \cdot \rangle$ is the inner product and $\|\cdot\|$ is the L^2 norm.

5.1.2 Datasets

Chord recognition datasets

Audio collections with the corresponding ground-truth labels of high quality have always been an essential condition for any MIR system assessment. Th proposed approach to chord recognition described in the previous chapter includes training block, which is necessary to perform model parameter estimation. This fact requires the dataset to be split into training and test parts. Here we utilize standard N-fold cross validation approach, where all the data is divided into N parts. Evaluation procedure is executed N times, each time one part is used as test material, while the rest of the collection is used for training purposes. Our evaluation setup, similarly to MIREX[1], performs 3-fold or 5-fold cross-validation, which means that all the songs were randomly divided into three or five folds.

[1] http://www.music-ir.org/mirex/2010/index.php/Main_Page

Data collections used for evaluation consist of the commonly used Beatles data set and additional 45 songs of Queen, Zweieck, and Carol King. The corresponding labels were kindly provided by C. Harte[79] and M. Mauch [27]. Two datasets are introduced. The first one, which will be referred to as "Beatles dataset", consists of 180 Beatles songs. Album names are given in Table A.1. The second one, which will be referred to as "Beatles-Queen dataset" consists of the "Beatles dataset" enriched with the songs of Queen, Zweieck, and Carol King. Songs of Queen, Zweieck, and Carol King are listed in Table A.2.

Chroma quality evaluation datasets

For this set of evaluations, we used a large set of recordings of individual notes collected at the University of Iowa [2]. This dataset contains high-quality note samples recorded from different instruments.

We used this data for generating chord waveforms. For a given chord type, the recordings of three constituent notes are chosen from three random instruments. Then these samples are mixed together, producing a waveform of 2 seconds duration. The proposed schema of generating data results in 200 waveforms with the corresponding ground-truth information on the notes.

The obtained material is then used to evaluate the quality of different chroma features as was described in section 5.1.1. For the RR measure, half of the generated material was used as training set, the other half was used for testing purposes.

5.2 Baseline configuration: impact of tuning

In this section, we evaluate the baseline configuration of our system. This configuration exploits standard chroma features that were introduced in Section 3.1. "No-LM" system configuration described in Section 4.3.3 is investigated. It allows us to assess the performance of the chosen feature set and evaluate the effectiveness of the proposed acoustic modeling.

5.2.1 Results

The first set of experiments considers different window lengths. Varying insertion penalty allows for obtaining output labels with different degree of fragmentation. The recognition accuracy as a function of insertion penalty, introduced in Section 4.1, for Hanning window is displayed in figure 5.1. For each window size, there is an optimal value of insertion penalty, which produces labels with a fragmentation rate very close to the ground-truth. Fragmenta-

[2] http://theremin.music.uiowa.edu/MIS.html

tion rate is another important characteristic of the transcribed chord labels, which is defined as relative number of chord labels [30].

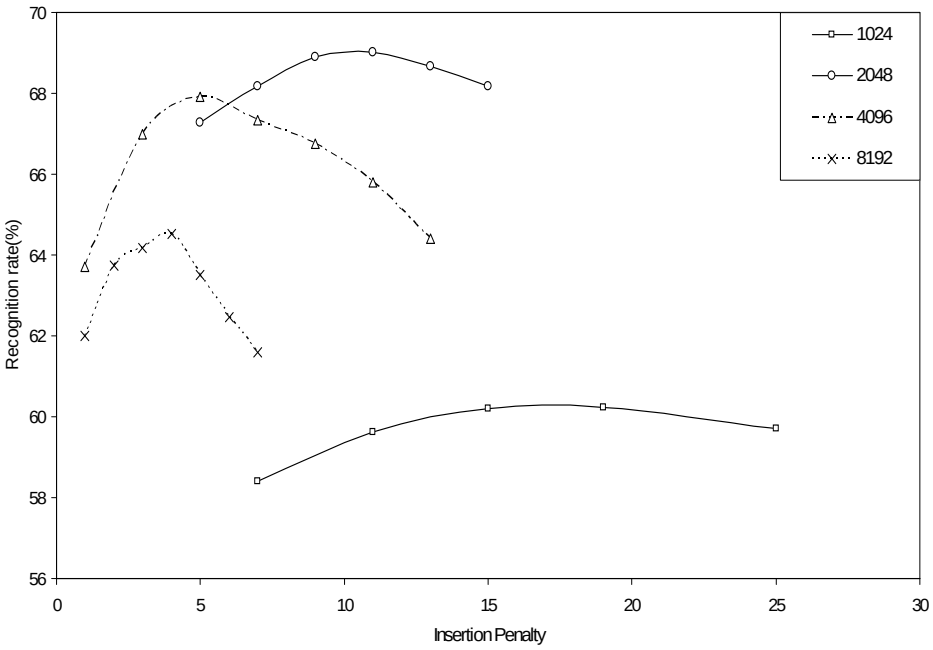


Figure 5.1: Recognition rate as a function of insertion penalty using Hanning window of different lengths.

In order to find the best windowing function, a set of tests were carried out involving window lengths of 1024(92.8 ms), 2048(185.7 ms), 4096(371.5 ms), 8192(743.0 ms), for Blackman, Hamming and Hanning window types (with 50% overlapping and the optimal insertion penalty). The results for the first fold are reported in table 5.1.

	1024	2048	4096	8192
Blackman	57.05	68.92	68.67	64.36
Hamming	60.24	69.00	67.91	64.18
Hanning	59.76	68.51	68.40	63.63

Table 5.1: System performance obtained with different windowing configurations on the first fold.

The highest performance (69.00 %) was achieved with Hamming window of length 2048 samples, while other window types showed results that are very close to this value. Window length of 2048 samples appeared to be a reasonable trade-off between the stationarity of the analysed frame of signal and frequency resolution. Taking the best configuration from the above-described experiments (Hamming window of length 2048 samples) the system perfor-

mance was conducted by including the tuning procedure. Different window delays D were explored in terms of recognition rate. The results are given in Table 5.2. By increasing the delay D, a very small increase in accuracy can be noticed, which can be due to a different uncertainty in frequency that is obtained for the given window length [80]. Besides this aspect, applying the tuning procedure leads to a higher recognition rate.

delay (samples)	recognition rate
1	71.37
2	71.42
4	71.41
10	71.52
12	70.60
15	69.06

Table 5.2: Recognition rate obtained using the tuning procedure.

In order to estimate the increase of performance introduced by the tuning procedure, a 3-fold cross-validation was accomplished on the "Beatles" data set. The results are shown in table 5.3, which show that about 2.5% and 1% improvements are obtained on the reduced and on the whole data sets, respectively.

data	baseline		with tuning	
	rec.rate	frag.	rec.rate	frag.
fold1	69.00%	0.80	71.52%	0.81
fold1, fold2, fold3	67.47%	0.84	68.28%	0.84

Table 5.3: Recognition rates and fragmentation rates on the reduced and on the complete test data set.

5.2.2 Conclusion

In this section, the results of a set of chord recognition experiments have been outlined which are based on exploring different windowing solutions as well as on the adoption of a tuning procedure to make this task less dependent on possible instrument mis-tuning effects. A novel approach for tuning introduced in section 3.2 that is based on concurrent analyzing magnitude and phase-change spectrum proved to be effective. Experimental results showed an increase in performance using the database of Beatles songs, for which an average recognition rate of 68.28% has been obtained.

5.3 Factored language models

In this set of experiments we evaluate "LM" and "FLM" system configurations introduced in Section 4.3.3. 5-fold cross-validation on the "Beatles" dataset is adopted. The use of standard and factored 3-gram and 4-gram language models is investigated. While working with FLMs, we exploited standard and generalized parallel back-off strategies (see Figure 4.4; 4-gram graphs have the same structure and can be obtained from 3-gram graphs by adding one level).

5.3.1 Results

Applying different language model weights on the stage of lattice rescoring, one can obtain different recognition rates. Figure 5.2 indicates how recognition rate depends on the LM weight. In this case, the curves correspond to the "LM" and "FLM" system configurations; experiments were conducted on the fold 1 with 4-gram configuration.

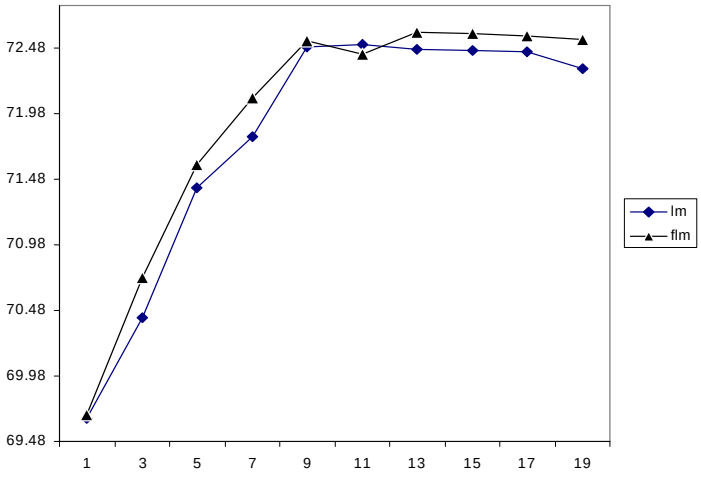


Figure 5.2: Recognition rate as a function of LM weight.

The recognition rates are shown in Table 5.4. Here "No-LM" is the baseline system, "3lm" "3flm" "3flmgpb" are trigram configurations for standard LM, FLM, and FLM with generalized parallel back-off respectively, "4lm" "4flm" "4flmgpb" are corresponding 4-gram configurations. For any of the given configurations, an average standard deviation of about 15% was also observed, which was derived from the recognition rates computed on a song-by-song basis.

Experimental results showed that introducing language modeling increases the performance of the system, while generalized parallel back-off strategy for FLM did not show any advantages over standard back-off for the chord recognition task. Meanwhile, using FLM show very slight

data	No-LM	3lm	3flm	3flmgpb	4lm	4flm	4flmgpb
fold 1	70.81	72.22	72.55	72.56	72.39	72.53	72.27
fold 2	70.23	70.78	71.15	71.51	71.09	71.38	71.25
fold 3	65.87	66.81	66.59	67.01	67.22	66.89	67.17
fold 4	66.20	67.15	67.60	67.61	67.64	67.62	67.51
fold 5	66.19	69.73	69.72	68.55	68.55	69.72	69.77
average	67.86	69.34	69.52	69.45	69.38	69.63	69.59

Table 5.4: Recognition rates for "No-LM", "LM", and "FLM" configurations.

improvement (0.25 %) in comparison with the standard LM.

5.3.2 Conclusions

In this section a set of experiments on chord recognition task including language modeling functionality as a separate layer has been conducted. The experimental results in a 5-fold cross-validation were conducted on the "Beatles" dataset. Factored language models were compared with standard language models and showed small increase in performance for the task. Comparing back-off techniques, we can assume that using generalized parallel back-off for the chord recognition task does not result in better performance.

In general, experimental results showed that utilizing language models leads to an increase in accuracy by about 2%. This relatively small difference in performance may be due to the size of vocabulary for the chord recognition task in comparison with that of many speech recognition applications. The performance of chord recognition systems is perhaps influenced primarily by relevance and accuracy of the extracted features and related acoustic modeling. A deeper study on different model smoothing and selection techniques as those addressed by Scholz et al. [47] could be reprised in further investigations.

5.4 PQMF-based chroma features

The next set of experiments investigates the performance of PQMF-based chroma features described in Section 3.3. Evaluation was performed in 3-fold cross-validation fashion on the "Beatles" dataset. "LM" system configuration is adopted here, where language models are represented by standard 3-grams.

5.4.1 Results

In order to compare the proposed PQMF-based chroma feature extraction technique with the standard one, a set of experiments with standard chroma were carried out. Frame lengths used in the experiments for the standard chroma were 185.76, 92.88 and 46.44 ms with the overlapping factor of 50%. For the PQMF-based chroma no overlapping was used. In order to operate at the same frame rate, the corresponding frame lengths of 92.88, 46.44 and 23.22 ms were used.

Figure 5.3 depicts the recognition rates of standard and PQMF-based chroma configurations as a function of insertion penalty. Different curves in the graphs correspond to different number of Gaussians in the mixtures for modeling emission probabilities in the HMMs.

Table 5.5 shows the evaluation results. The recognition rates in each row are the best among possible configurations (penalty, number of Gaussians) for a specified frame length.

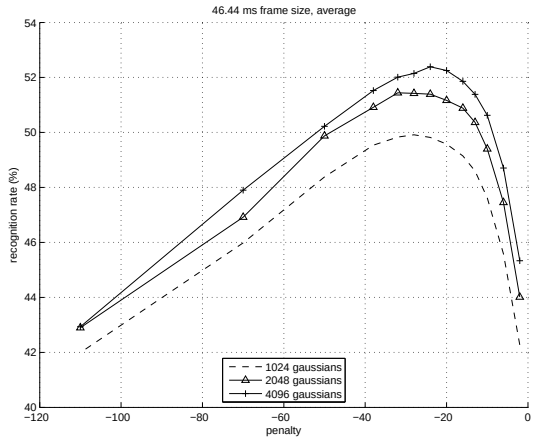
	frame size (ms)	best result (%)
PQMF chroma	23.22	69.37
PQMF chroma	46.44	69.43
PQMF chroma	92.88	68.31
Standard chroma	46.44 (50% overlap)	52.39
Standard chroma	92.88 (50% overlap)	64.53
Standard chroma	185.76 (50% overlap)	69.53

Table 5.5: Evaluation result summary. Best recognition rates for different frame lengths and feature extraction methods.

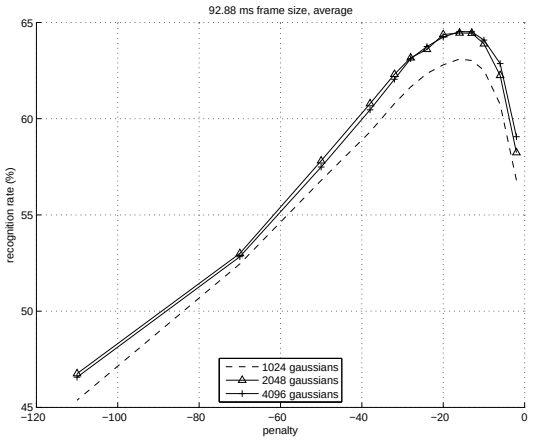
5.4.2 Conclusions

The experimental results show that chroma extraction based on PQMF filter bank analysis and subsequent periodicity detection does not outperform the standard approach for the analysis frame length of 182 ms. However, when taking into consideration short-term analysis with frame lengths of 46 ms and 92 ms the proposed approach significantly outperforms the application of standard chroma feature vector extraction. The proposed technique could be of great use in the music transcription tasks where it is necessary to capture harmonic content of the signal with very high time resolution. To this end, new specific tasks will also be devised in the future activities.

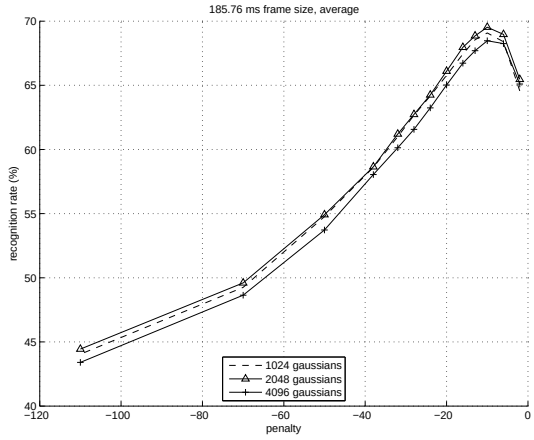
One of the main disadvantages of the filter bank approach can be the very high computational load if compared to the standard chroma extraction. Although, the issue of complexity will be subject of future investigation. In spite of the fact that each filter has passband bandwidth



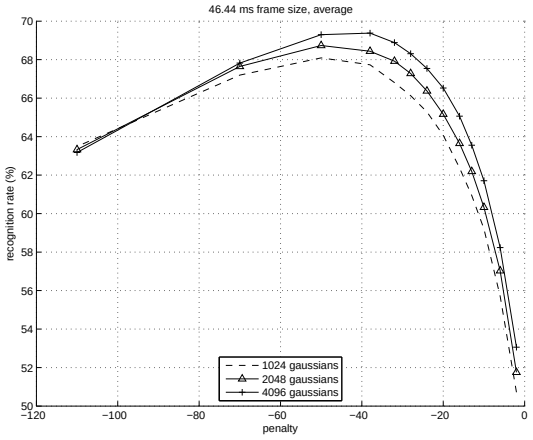
(a) Standard chroma with frame length of 46 ms.



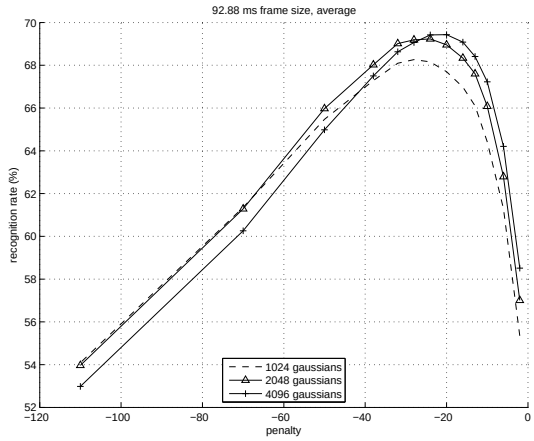
(b) Standard chroma with frame length of 92 ms.



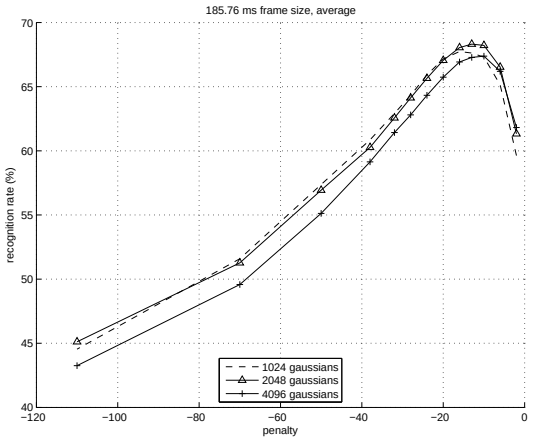
(c) Standard chroma with frame length of 185 ms.



(d) PQMF-based chroma with frame length of 46 ms.



(e) PQMF-based chroma with frame length of 92 ms.



(f) PQMF-based chroma with frame length of 185 ms.

Figure 5.3: Recognition rates for different system configurations as a function of insertion penalty.

wider than semitone distance, PQMF-based chroma features showed comparable performance to the standard chroma features.

It is worth noting that alternative filter bank configurations (e.g. with more channels) can be utilized. In the above-described configuration, the width of each channel of the filter bank is approximately equal to 2 – 3 units of the semitone distance. However, introducing filters with narrower bandwidth can cause the substantial increase in the lengths of the filters, therefore causing a further increase of the computational load.

Another wide area of research may regard different alternative techniques for the periodicity detection. In fact, periodicities computed in the previous frames can be exploited for a more effective computation of the periodicity in the actual frame.

5.5 Time-frequency reassigned chroma features

In this section, we carry out experiments with TFR-based chroma features. Similarly to the previous experimental setup, "LM" system configuration with standard 3-grams is adopted here. "Beatles-Queen" collection is utilized as the evaluation dataset.

5.5.1 Chroma quality evaluation

Initial set of experiments is aimed at comparing standard chroma feature with the RC and HRC features introduced in Section 3.4. Chroma quality evaluation was performed using the metrics described in Section 5.1.1. Chroma features were extracted with 185 ms window lengths, overlapping factor of 90% and Hanning windowing. The evaluation results for three different chroma features are given in Figure 5.4.

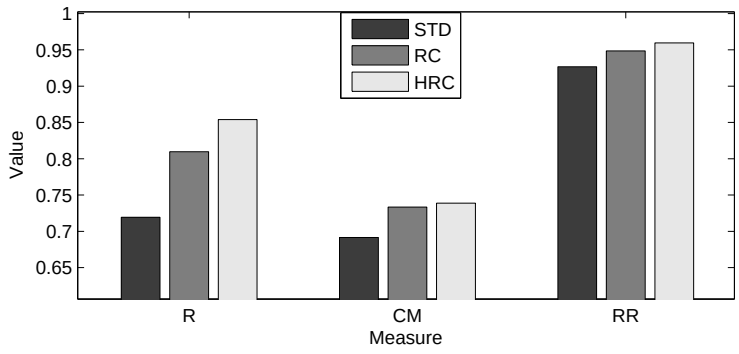


Figure 5.4: Chroma quality estimates.

In all the cases, *HRC* and *RC* significantly outperform *STD* feature. In particular, the *ratio* measurements proved the ability of *HRC* to deemphasize noise and impulsive components, which frequently occur during the note onsets.

5.5.2 Chord recognition system evaluation

Baseline system configuration

As a baseline system we define a single-stream HMM with standard chroma feature and "LM" decoder. Previous experiments provided in Section 5.2 showed that optimal window length for standard chroma is 185 ms. As starting point, we observe 70.62% recognition rate on the "Beatles-Queen" dataset. In the following sections different window lengths will be used for the STD feature leading to different results. The baseline system configuration does not contain tuning block. Tuning issues will be addressed later in this section.

Time-frequency reassigned chroma features with reassignment constraints

In this set of experiments, we introduce the RC feature set and investigate its behavior applying reassignment constraints. In order to estimate the impact of the time-frequency reassignment operation, statistical information on the energy reallocation distance in time-frequency coordinates has been collected.

For window length of 96 ms, Δf and Δt distributions can be approximated with a Gaussian with zero-mean and standard deviations of 15.68 Hz and 143 ms respectively.

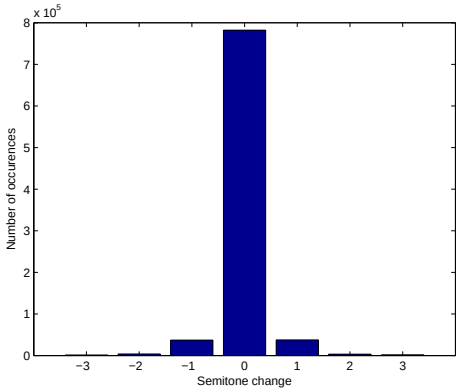


Figure 5.5: Semitone change distribution

Frequency shift (in semitones)	-3	-2	-1	0	1	2	3
Percentage of energy reassignment	0.14	0.43	4.28	90.23	4.33	0.37	0.19

Table 5.6: Semitone change distribution

The statistics about the frequency reassignments that lead to energy moving to another semitone bin is given in Table 5.6 and Figure 5.5. This table shows that about 9.7% of all the reassignments result in moving energy to an adjacent semitone bin, which makes an impact on

the chroma energy distribution. This preliminary statistical study shows the importance of the time-frequency reassignment operation. Reallocating a substantial amount of energy between different chroma components can improve the accuracy.

In order to estimate the efficiency of the time-frequency reassignment operation, other evaluation tests that impose reassignment restrictions were carried out. In these experiments, a time-frequency rectangular window is defined as shown in Figure 5.6 and all the energy reassignments are constrained to remain inside this window. Otherwise, the reassignment operation is not performed and original time-frequency coordinates are preserved. In practice, Δf and Δt are limited to small values for the energy reassignment shift to be allowed. Two examples of time-frequency rectangular window are given in Figure 5.6. In this schema, window width is represented by a maximum allowed reassignment in the time domain, and height is represented by that in the frequency domain.

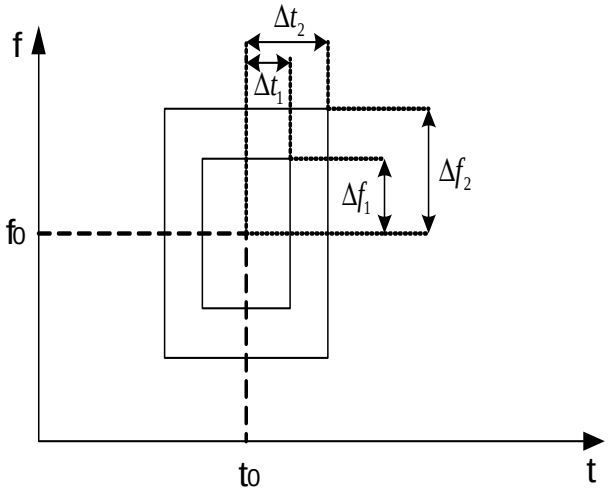


Figure 5.6: Schema of time-frequency reassignment window constraints

Experiments with different combinations of Δf and Δt were carried out. Figure 5.7 displays chord recognition rates applying various reassignment constraints Δf and Δt. Results showed that a minimum constraint of 100Hz-1sec is necessary to approach the performance provided by the unconstrained reassignment with the best recognition rate. The difference between the proposed TFR-based features and standard ones turned out to be about 6%.

Some results from Figure 5.7 are given in Table 5.7. For this set of results, a Tukey's HSD test was also run. Figure 5.8 proves the fact that enlarging Δf and Δt results in system configurations with statistically significant differences in chord recognition rates.

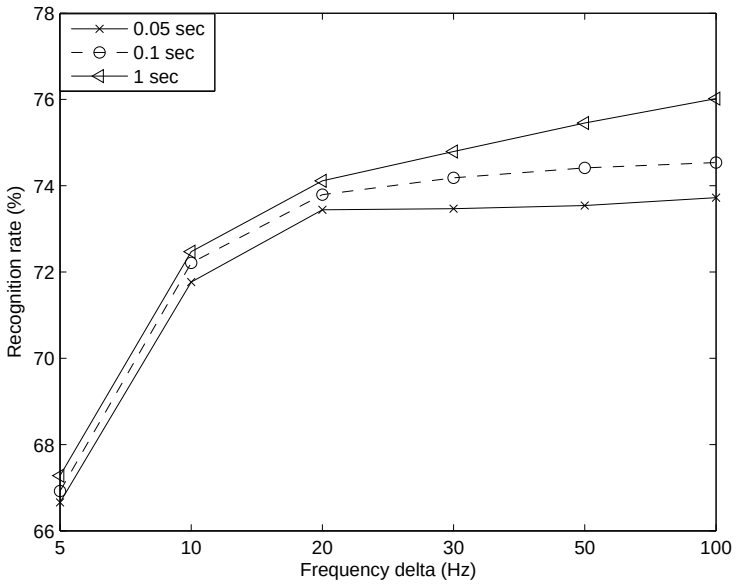


Figure 5.7: Evaluation results with time-frequency reassignment constraints as a function of Δf. Different Δt are represented by different curves.

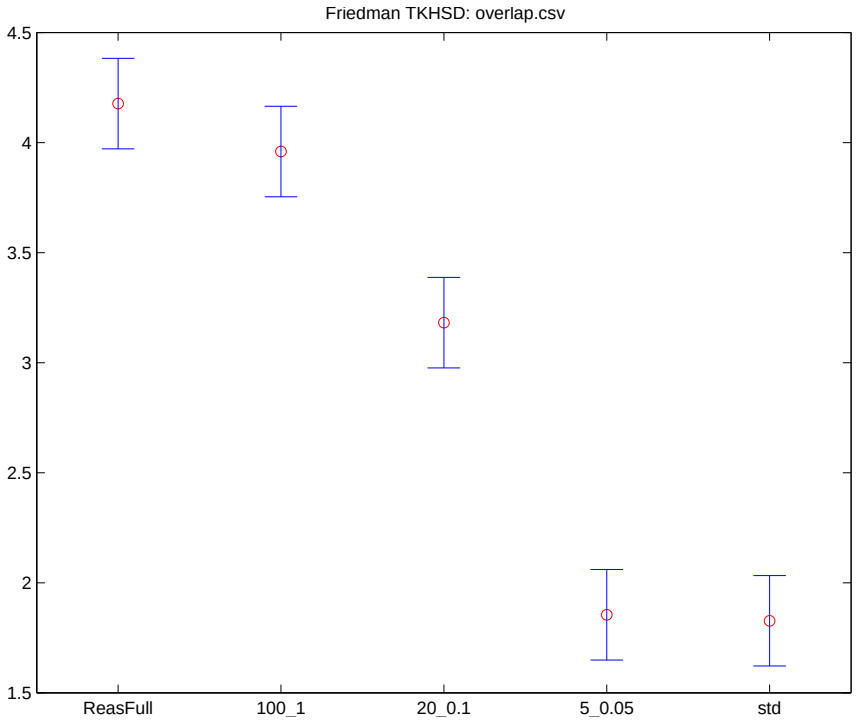


Figure 5.8: Tukey's HSD test.

std	5_0.05	20_0.1	100_1	reas
65.38	66.65	73.79	76.02	76.70

Table 5.7: A subset of evaluation results with time-frequency reassignment constraints.

Alternative settings

Since the time-frequency reassignment technique used here includes weighting an analyzed signal with a window function, an impact of different window types on the performance was investigated. The results for the STD (using 192 ms window length) and RC (using 96 ms window length) feature are provided in Table 5.8.

From these results, it is shown that system performance does not vary greatly with different window types. Blackman, Hanning and Hamming windows showed quite similar results. Similar behavior was observed in the experiments described in Section 5.2.

	STD	RC
hanning	70.62	76.70
hamming	70.5	76.63
blackman	70.41	76.56
kaiser(alpha=8)	70.36	76.82

Table 5.8: Performance of STD and RC feature with different window types.

A number of different configurations is involved for optimizing such parameters as spectrum type (energy or magnitude), number of Gaussians to model emission probabilities and insertion penalty.

Figure 5.9 depicts recognition rates using RC feature with different window lengths and number of Gaussians. Hanning window is assumed here and later on. For each configuration the best insertion penalty is assumed.

These results showed that for the RC feature optimal window length appeared to be 96 ms, as opposed to the STD feature, for which such a short window length results in a much lower performance. This fact is coherent with a more accurate energy localization in time for the TFR-based features.

Figure 5.10 presents further investigation on the impact of the spectral energy rate δ introduced in Equation (3.16). In the case of magnitude spectrum δ value is set to 1, for power spectrum it is set to 2. An optimal parameter setting from the previous experiments is here assumed (RC feature, 96 ms Hanning window). The optimal value for the given dataset and approach is around 0.75.

An important step in the feature extraction process is the estimation of the deviation of the

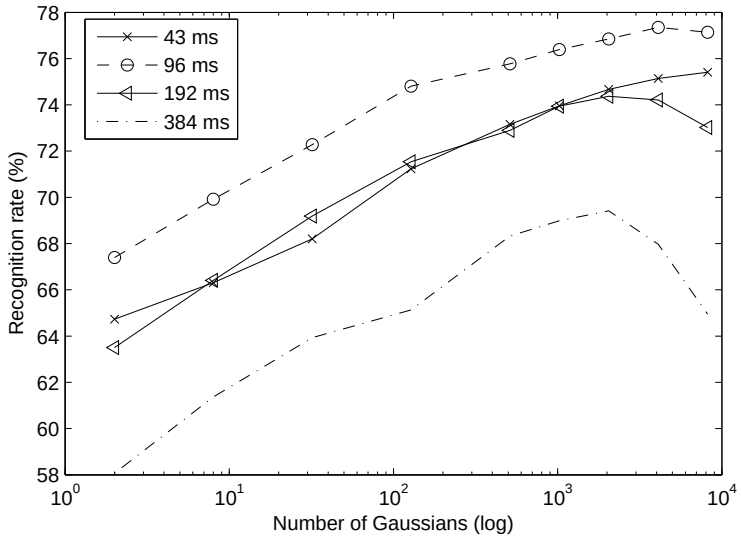


Figure 5.9: Recognition rates using the RC features for different window lengths and Gaussian numbers

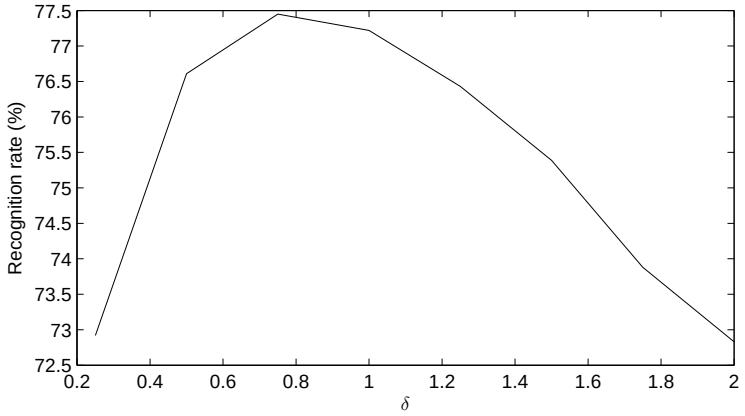


Figure 5.10: Recognition rate for RC feature as a function of δ.

A4 note from 440 Hz and subsequent compensation for mis-tuning [67]. Since a considerable amount of data used for the evaluation purposes was recorded several decades ago, the tuning problem should be taken into careful consideration. Here we reprise the experiments that show the impact of tuning block. Results with the STD and RC features with and without tuning are provided in table 5.9.

STD	STD tuning	RC	RC tuning
70.33	71.29	76.70	77.29

Table 5.9: Influence of tuning on STD and RC feature performance

The experimental results showed that the tuning operation plays an important role and leads to an increase in performance of about 0.6% for the RC feature, similarly to what was observed in Section 5.2.

A large-scale parameter optimization performed here lead to interesting results. Different window types showed similar performance, RC feature showed the best results with 96 ms window length. Taking magnitude spectrum instead of energy leads to better performance. Moreover, using δ value of 0.75 leads even to a better performance. The usage of tuning block proved to be reasonable.

Harmonic reassigned chroma

In order to improve the quality and robustness of the RC feature, and take an advantage of possible harmonic filtering of the reassigned spectrogram introduced in section 3.4, the adoption of the HRC features is here explored.

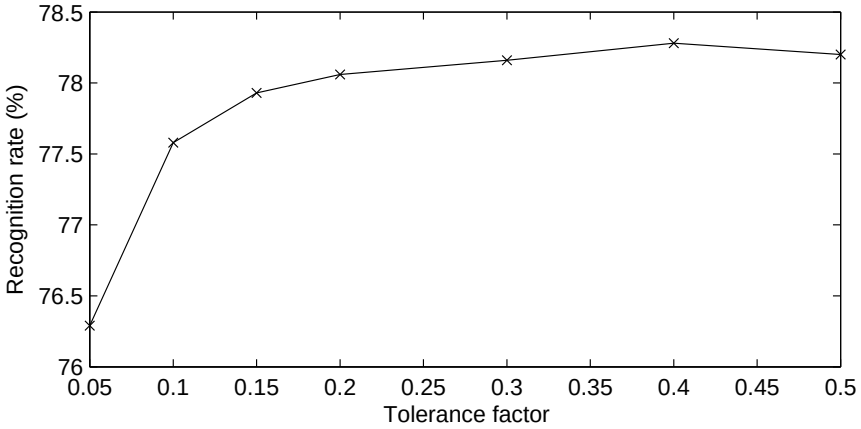


Figure 5.11: Recognition rate for HRC as a function of the tolerance factor

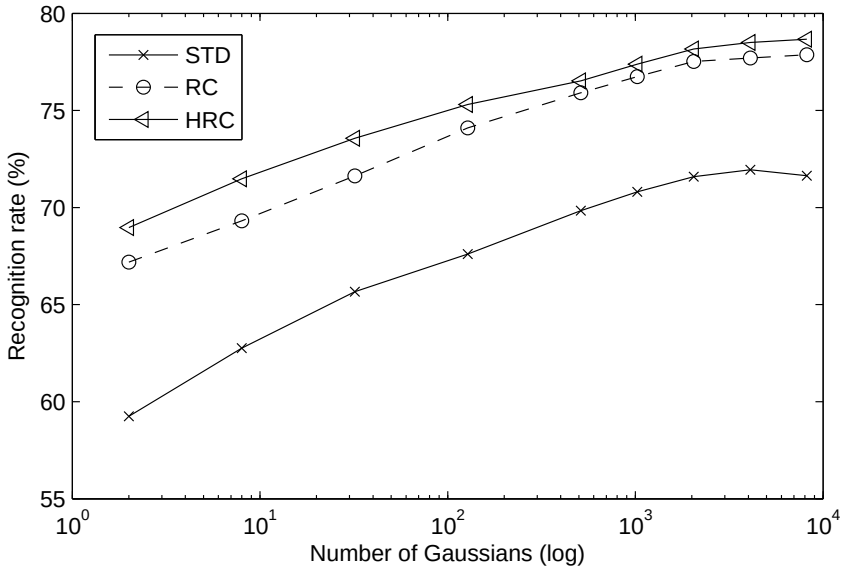


Figure 5.12: Recognition rate as a function of the number of Gaussians

First of all, the impact of the tolerance factor A introduced in (3.23) was investigated. As shown in Figure 5.11. The optimal value of A for the chord recognition task turned out to be 0.4 with recognition rate of 78.28%, although small deviations on this parameter seem to have a minor impact in terms of loss of performance.

The next set of experiments aimed to compare HRC, RC and STD features. Figure 5.12 depicts recognition rates for different number of Gaussians. For each configuration the best insertion penalty is assumed.

In all the three cases, the obtained results indicate good choice of the number of Gaussians equal to 2048. Higher values do not bring significant improvement, while increasing computational load drastically. This trend may also depend on the training material size. As a result, the HRC feature proved to be advantageous over RC with the optimal value of A to be 0.4 with chord recognition rate improved to 78.28%.

Chroma and bass-chroma in multi-stream HMMs

Having shown the advantage of the HRC features, in the next sections we will adopt them for further investigations. The next step is based on the models with multi-stream observation layer introduced in section 4.1. This set of experiments involved the technique of splitting frequency range used for chroma calculation into two parts: chroma and bass-chroma. For computing bass-chroma, frequencies that correspond to the MIDI range between 24 (32.7 Hz) and 54 (185 Hz) notes are used. For chroma feature extraction frequency interval between 54 (185 Hz) and

96 (2093 Hz) MIDI notes is employed. For these experiments, we used HRC feature based on the spectrum calculated with Hanning window of 92 ms and tolerance factor set to 0.4. The obtained chord recognition rate turned out to be equal to 80.26%, i.e., multi-stream HMMs provided a further improvement of about 2%.

Thus, this bunch of experimental results proved the fact that splitting frequency region into 2 bands is reasonable and leads to a significant increase of chord recognition rate.

Chroma and bass-chroma weights

In order to take further advantage of using the two chroma streams, a careful evaluation of the system performance was performed setting different stream weights in the Viterbi recognizer.

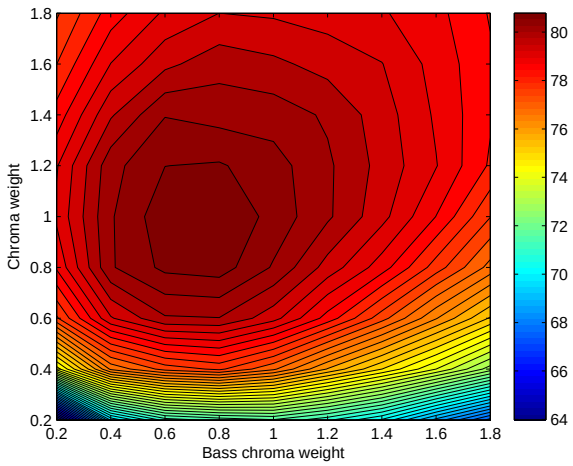


Figure 5.13: Recognition rate (%) as a function of different weights for chroma and bass-chroma observable streams

Important parameters here are the weights of each stream. Figure 5.13 depicts recognition rate as a function of bass-chroma weight and chroma weight. The self-test experiments, when the training material was used as a test set, were also conducted. The obtained results, shown in Figure 5.14, suggested the optimal stream weights for the given data corpus.

The experiments of this section proved that assigning different importance factors to different feature streams by applying stream weights in the recognizer is effective. It was shown that using weights 1, 0.7, for the chroma and bass-chroma streams, respectively, leads to the best performance of 81.58%.

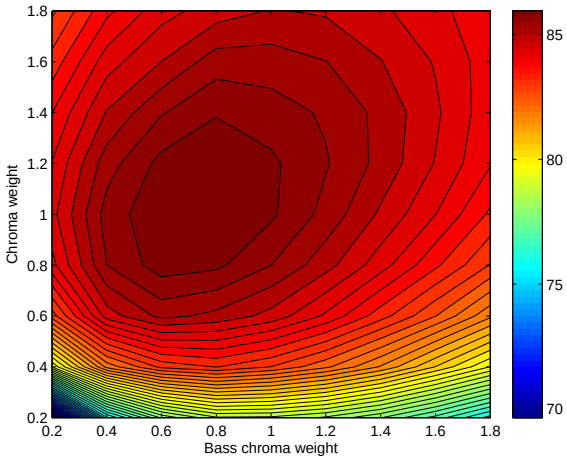


Figure 5.14: Self-test recognition rate (%) as a function of different weights for chroma and bass-chroma observable streams

Number of emitting states in HMM

A series of experiments involving more than 1 emitting states was conducted. Table 5.10 provides the summary of recognition rates as a function of Gaussian number for different number of states (1 - 3). using 2 or 3 emitting states in HMM does not bring any improvement.

	2	8	32	128	512	2048
1 state	72.6	75.15	77	78.89	79.91	81.58
2 states	71.96	74.88	77.6	78.55	79.74	81.22
3 states	71.93	74.59	77.22	79.03	79.87	80.82

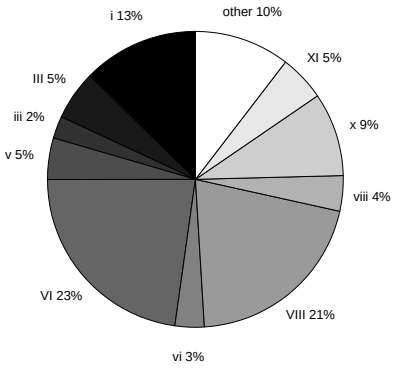
Table 5.10: Recognition rates as a function of Gaussian number for different number of states in HMM

Chord confusions

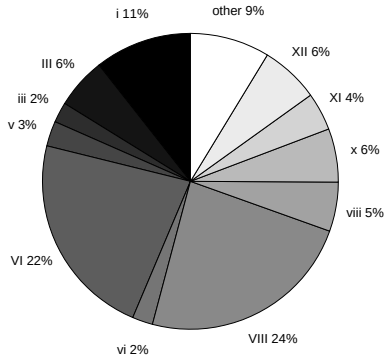
Finally, in order to understand in detail chord misclassifications statistics, information about typical errors was collected. The confusion pie charts for the baseline and best system configurations are presented in figures 5.15.

The relation between detected chord and ground-truth chord is denoted by Roman numerals. Lower-case numerals are used to indicate minor triads and upper-case for major ones. For example, wrongly detected **G** instead of **C** is indicated as **VI**.

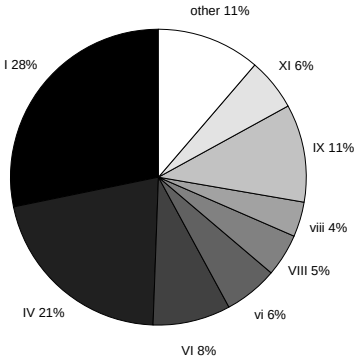
Major chord confusions of the baseline and the best system configurations do not show any significant difference of the error statistics. At the same time, number of "parallel" errors for



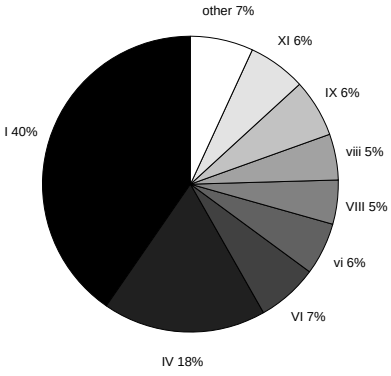
(a) Baseline for *major* chord



(b) Best configuration for *major* chord



(c) Baseline for *minor* chord



(d) Best configuration for *minor* chord

Figure 5.15: Chord confusion statistics.

Minor increases to a considerable extent from 28% in the baseline system to 40% in the final configuration. Significantly reduced the number of **XI** confusions from 11% to 6%. However, other error type statistics are similar for the baseline and best system configurations.

Conclusions

In this section we proved the fact that accurate spectral analysis for feature extraction can significantly improve chord recognition accuracy. RC feature performed significantly better than standard chroma. Large-scale evaluations of chord recognition system with different parameter configurations pointed out the optimal feature, which is HRC chroma with tolerance factor set to 0.4. Multi-stream HMM configuration, where the two observable streams correspond to chroma and bass-chroma proved to be effective and showed better performance in comparison with a single-stream HMM configuration. A substantial improvement over the baseline system has been obtained with the final result of 81.58% recognition rate.

5.6 Chroma features with reduced dimensionality

This section is concerned with the evaluation of chroma features with reduced dimensionality introduced in Section 3.5. Tonal centroid as well as different IDCT features are evaluated. Experimental setup used here is the same as described in Section 5.5.2.

The experimental results are given in Table 5.11. Here "2 chroma streams" is the best configuration obtained using bass and treble chroma streams with the corresponding stream weights of 0.7 and 1.0. "Unique chroma vector" configuration utilizes c_{com1} vector as feature set. "2 tonal centroid streams" is the 2-stream configuration with bass and treble tonal centroids weighted by 1.0 and 1.0 correspondingly. In "Tonal centroid treble" and "Chroma treble" we investigate the advantages of tonal centroid over standard chroma. And finally "IDCT c_{com1}" and "IDCT c_{com1}" are the IDCT features given in (3.29) and (3.30), while "IDCT c_{com1} subtract mean" shows system performance using mean subtraction technique.

Experimental results showed that tonal centroid did not show any advantage over standard chroma features, neither in a single-stream, nor in a multi-stream configuration. Using IDCT transform that is considered to be established technique in speech processing did not prove its effectiveness in chord recognition. Mean subtraction did not show any advantages, conversely, it proved to decrease the performance drastically.

Configuration	Dimensionality	Recognition rate
2 chroma streams	12+12	81.581
Unique chroma vector	24	79.339
2 tonal centroid streams	6+6	79.753
Tonal centroid treble	6	74.408
Chroma treble	12	76.691
IDCT c_{com1}	16	77.624
IDCT c_{com1}	16	77.211
IDCT c_{com1} subtract mean	16	71.573

Table 5.11: Experimental results using feature dimensionality reduction

5.7 MIREX evaluations

In this section we present the results of the proposed chord recognition systems that participated in MIREX competitions. We compare the performance with other submitted systems and analyze statistically significant differences in the results.

5.7.1 MIREX 2008

The first time audio chord recognition was included in the list of MIREX subtasks was in 2008.

At that time several approaches to chord recognition existed, but comparison of the output results was difficult, because different measures were used to assess the performance. MIREX 2008 established common rules and methodology for chord recognition systems evaluation. Test set, which included 176 songs of Beatles, was defined. At that time it was the largest and probably the only publicly available labeling dataset of ground-truth chords kindly provided by C. Harte. The audio was in WAV format in at a sampling rate of 44.1 kHz and a bit depth of 16 bit. Ground-truth to audio alignment was done automatically with the script provided by C. Harte. Audio chord detection task was divided into two subtasks, which are "train-test" and "pretrained". In the "pretrained" subtask participants were supposed to submit systems that are ready to perform chord transcription. All the parameters are set up in advance and no model training is needed. In the "train-test" subtask the process of system evaluation consisted it two steps. At first, model parameters are estimated using training data. In the last step, the trained system is evaluated on the test data. 3-fold cross validation was adopted, where album filtering was applied on each train-test fold. That means that songs from the same album can not appear in both train and test sets simultaneously.

Two different measures were used. The first measure, that was called "Overlap score", is the "recognition rate" measure introduced in Section 5.1.1. It is calculated as ratio between the

duration of correctly identified chords and total duration of ground-truth chords. The second measure, that is "Overlap score merged" is calculated in a similar manner. The only difference is the fact that only chord roots from output labels are matched against ground-truth. For example, A:maj is considered to be correct if there is A:min in the ground-truth.

Participants from different teams are presented in Tables 5.12 and 5.13. The results for "pre-trained" and "train-test" subtasks are given in Figure 5.16. Tukey-Kramer HSD tests for statistical significance are depicted in Figure 5.17.

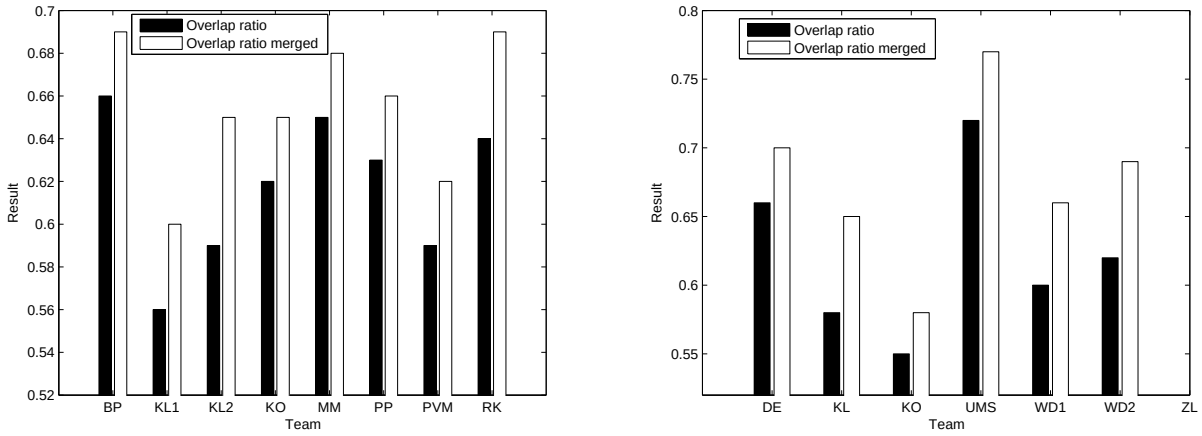
Team ID	Authors
BP	J. P. Bello, J. Pickens
KO	M. Khadkevich, M. Omologo
KL1	K. Lee 1
KL2	K. Lee 2
MM	M. Mehnert
PP	H.Papadopoulos, G. Peeters
PVM	J. Pauwels, M. Varewyck, J-P. Martens
RK	M. RyynÃd'nen, A. Klapuri

Table 5.12: Team legend for MIREX 2008 pretrained subtask.

Team ID	Authors
DE	D. Ellis
ZL	X. Jhang, C. Lash
KO	M. Khadkevich, M. Omologo
KL	K. Lee
UMS	Y. Uchiyama, K. Miyamoto, S. Sagayama
WD1	J. Weil
WD2	J. Weil, J.-L. Durrieu

Table 5.13: Team legend for MIREX 2008 train-test subtask.

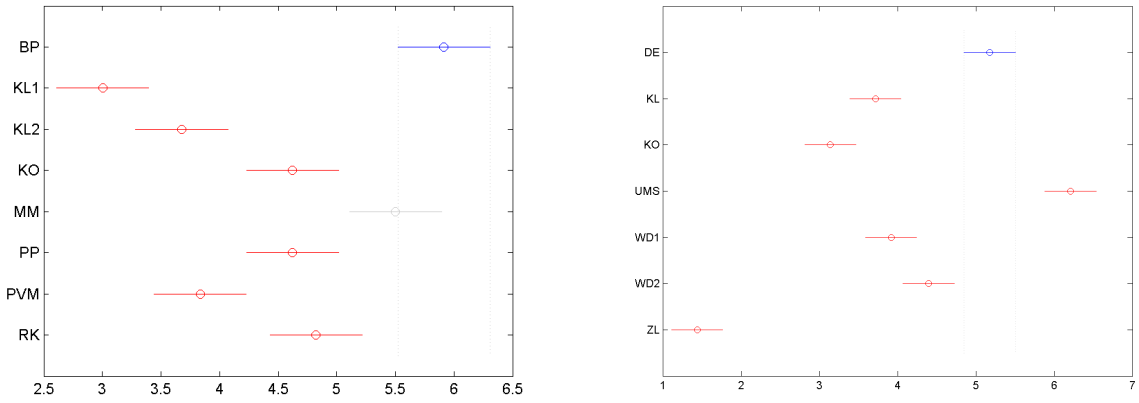
KO system that participated in both subtasks is described in Section 5.2 of experimental results. Standard chroma features that were introduced in Section 3.1 were used as a front-end. "No-LM" system configuration described in Section 4.3.3 was adopted. At that time, out chord recognition system did not include language modeling functionality. Parameter estimation for the "pretrained" system configuration was performed using the "Beatles" dataset. The difference in performance between our systems in "pretrained" and "train-test" subtasks appeared to be about 8%. It seems to be due to a small bug in the chroma computation module and the fact that "pretrained" system had seen the test material before, since it was previously used for model



(a) Pretrained subtask

(b) Train-test subtask

Figure 5.16: MIREX 2008 results in audio chord detection.



(a) Pretrained subtask

(b) Train-test subtask

Figure 5.17: Tukey-Kramer HSD test for MIREX 2008 results.

parameter estimation.

In the "pretrained" subtask the system of Bello and Pickens showed the best performance. However, the winner of the competition is undoubtedly the system of Uchiyama, Miyamoto, and Sagayama. It showed 72% overlap ratio with statistically significant difference from all other systems, as shown in Figure 5.17. The system of Ellis with the overlap score of 70% also showed the results that are significantly better than all the rest of the systems.

5.7.2 MIREX 2009

MIREX 2008 audio chord recognition contest had attracted the attention of many people working in the MIR area. 15 systems were submitted from 11 different researchers and research groups. Rules and methodology of evaluation in MIREX 2009 were derived from MIREX 2008. However, the test material was enlarged and comprised not only the Beatles collection, but 38 additional songs of Queen and Zweieck donated by Matthias Mauch.

Participants from different teams are presented in Tables 5.14 and 5.15. The results for "pre-trained" and "train-test" subtasks are given in Figure 5.18. Tukey-Kramer HSD tests for statistical significance are depicted in Figure 5.19.

Team ID	Authors
CH	C. Harte
DE	D. Ellis
KO1 – KO2	M. Khadkevich, M. Omologo
MD	Matthias Mauch, Katy Noland, Simon Dixon
OGF1 – OGF2	L. Oudre, C. Févotte, Y. Grenier
PP	H. Papadopoulos, G. Peeters
PVM1 – PVM2	Johan Pauwels, Matthias Varewyck,Jean-Pierre Martens
RRHS1 – RRHS3	T. Rocher, M. Robine, P. Hanna, R. Strandh

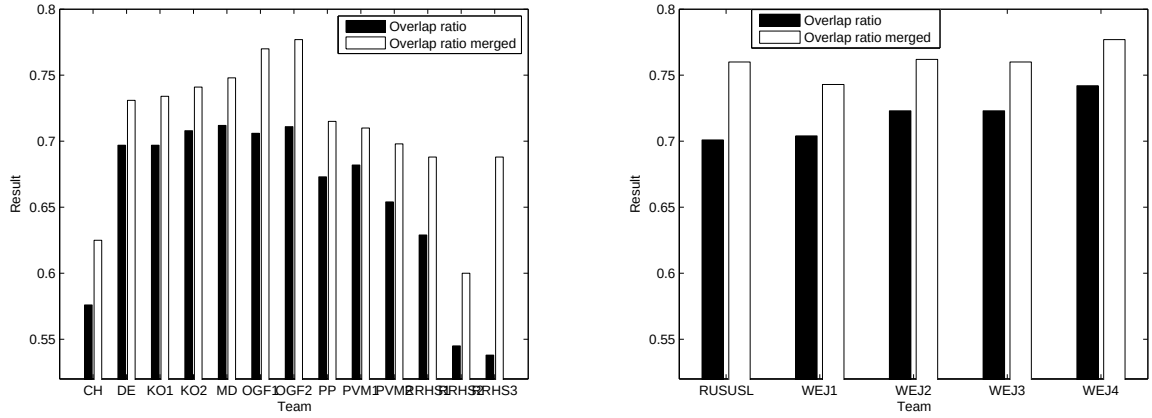
Table 5.14: Team legend for MIREX 2009 pretrained subtask.

Team ID	Authors
RUSUSL	J.T.Reed,Yushi Ueda,S.Siniscalchi,Yuki Uchiyama,Shigeki Sagayama,C.H.Lee
WEJ1 – WEJ4	Adrian Weller, Daniel Ellis, Tony Jebara

Table 5.15: Team legend for MIREX 2009 train-test subtask.

KO1 and **KO2** system was submitted to partcipate in the "pretrained" subtask. In comparison with the system **KO** that was submitted to MIREX 2008, several minor improvements in the feature extraction block were made. Mistuning rate estimator was added, which improved the front-end. **KO2** system was equipped with the language modeling block. The configuration is derived from the "LM" system described in Section 4.3.3.

Both systems, **KO1** and **KO2**, showed good results. The difference in overlap score between **KO2** and the best submission in the "pretrained" subtask, which is **MD**, appeared to be only 0.3%. The best system showed 71.2% of overlap score. The next result was produced by the system **OGF2** with the overlap score of 71.1%, which is extremely close to the highest result. There is no surprise that HSD test did not show significant differences between the bestsix re-



(a) Pretrained subtask

(b) Train-test subtask

Figure 5.18: MIREX 2009 results in audio chord detection.

sults, as shown in Figure 5.19. **KO1** and **KO2** systems showed 69.7% and 70.8% overlap ratios respectively. It is worth noting that in **KO1** no statistical information about chord transitions was used. Transition probabilities between each chord pair are equal and classification is based solely on acoustic features. Including language modeling in **KO2** showed a slight increase in performance.

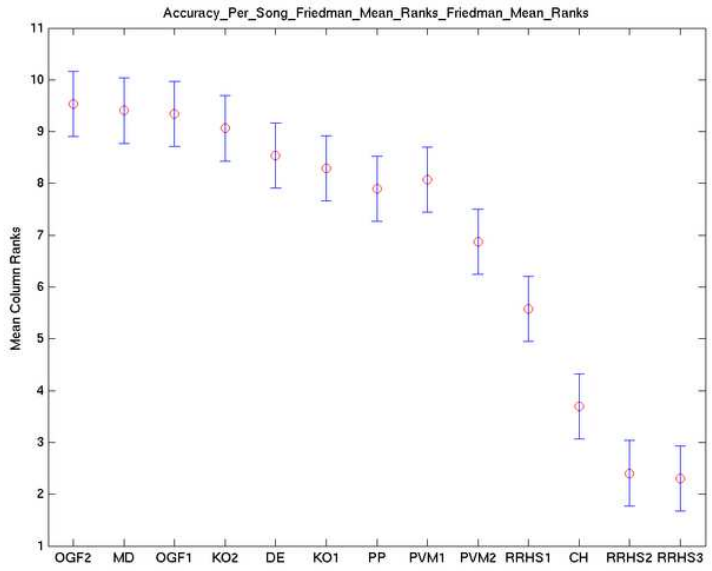
The leader in the "train-test" subtask is the submission **WEJ4** of Weller et al. with the overlap ratio of 74.2%. The algorithm is based on the application of SVM [81] and outperformed the best system from the "pretrained" subtask.

5.7.3 MIREX 2010

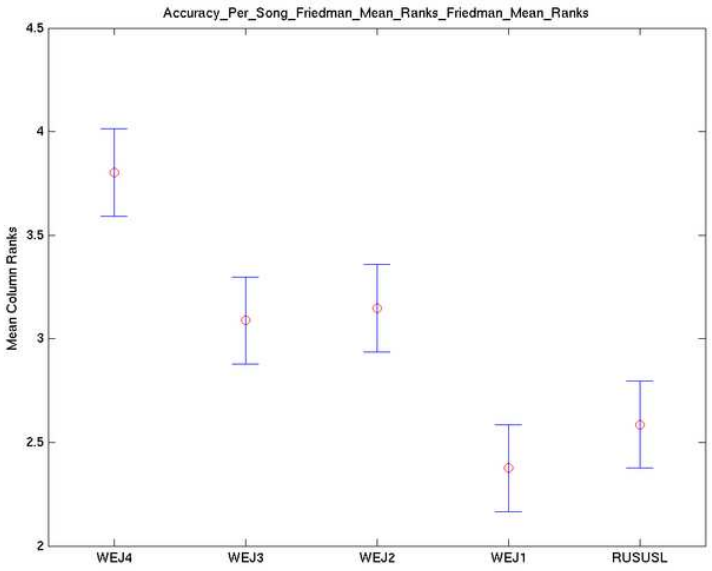
MIREX 2010 gave a new perspectives on large scale evaluation of MIR systems. NEMA MIREX DIY infrastructure was developed to facilitate the process of automatic processing the results. In contrast to the previous years, evaluation metrics changed. Instead of "overlap ratio merged", "weighted average overlap ratio" metric was introduced. "Weighted average overlap ratio" was calculated as the average overlap ratio calculated on the song basis. Dataset remained the same as in MIREX 2009. Starting from MIREX 2010 "pretrained" and "train-test" subtasks are merged together in a single "audio chord detection" task.

Participants from different teams are presented in Table 5.16. The results are given in Table 5.17. Tukey-Kramer HSD tests for statistical significance are depicted in Figure 5.20.

Two different systems were submitted. They are **KO1** and **MK1**. Recently developed RC features were used as the front-end. Multi-stream HMM were utilized for acoustic modeling, where frequency range for chroma calculation was split into two parts: chroma and bass-



(a) Pretrained subtask



(b) Train-test subtask

Figure 5.19: Tukey-Kramer HSD test for MIREX 2009 results.

chroma. "LM" system configuration was adopted. While **MK1** system needed training, **KO1** system was submitted with all the model parameters estimated in advance.

MIREX 2010 competition in chord detection showed significant increase in performance

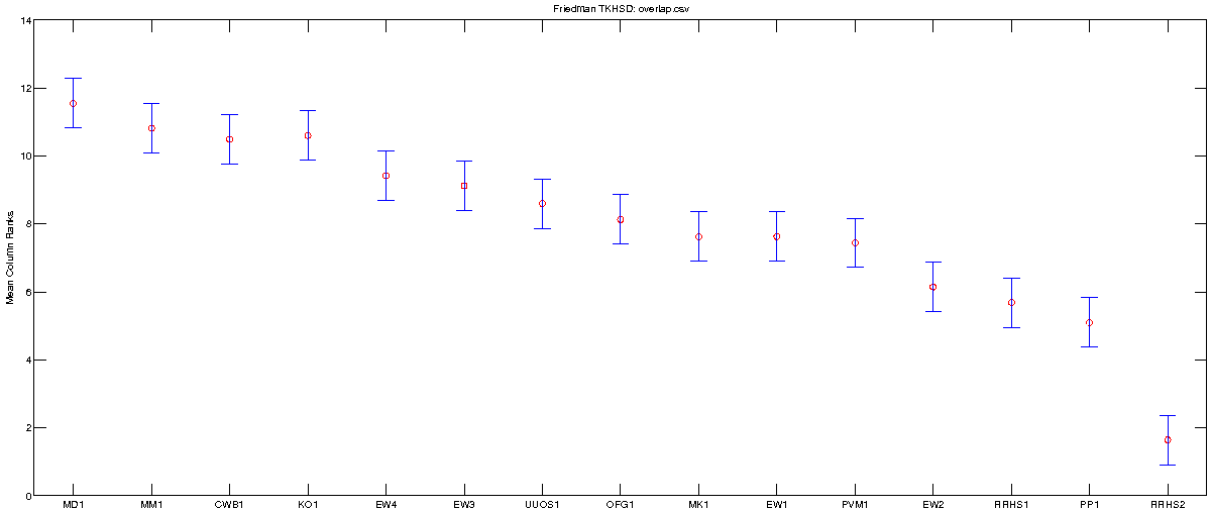
Team ID	Authors
CWB1	T. Cho, R. Weiss, J. Bello
EW1 – EW4	D. Ellis, A. Weller
KO1	M. Khadkevich, M. Omologo
MD1	M. Mauch, S. Dixon
MK1	M. Khadkevich, M. Omologo
MM1	M. Mauch
OFG1	L. Oudre, C. Févotte, Y. Grenier
PP1	H. Papadopoulos, G. Peeters
PVM1	J. Pauwels, M. Varewyck, J.-P. Martens
RRHS1 – RRHS2	T. Rocher, M. Robine, P. Hanna, R. Strandh
UUOS1	Y. Ueda, Y. Uchiyama, N. Ono, S. Sagayama

Table 5.16: Team legend for MIREX 2010 audio chord detection contest.

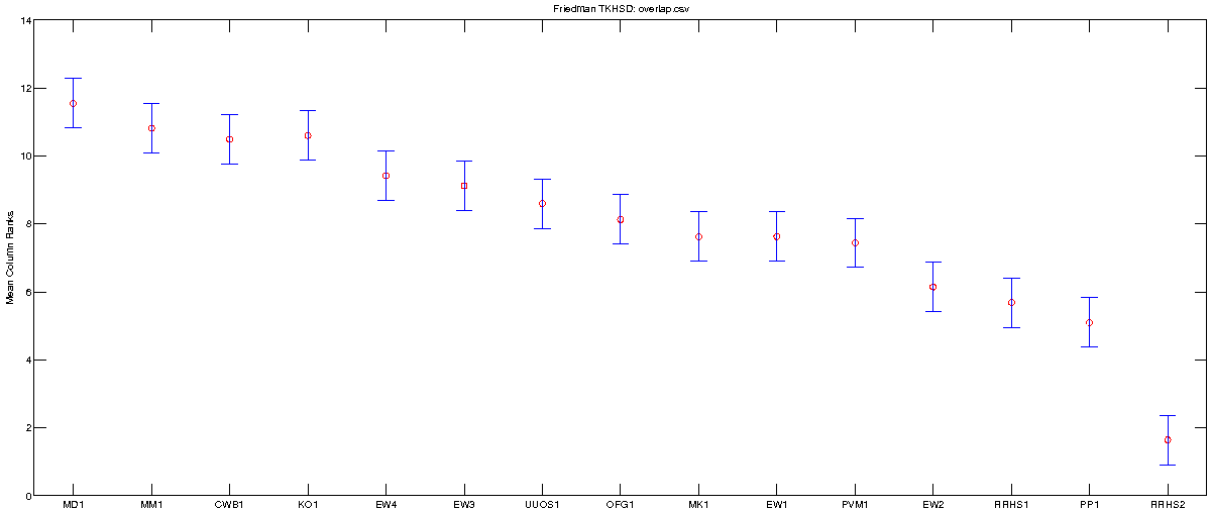
Algorithm	Chord Overlap ratio	Chord weighted average overlap ratio
MD1	0.8022	0.7945
MM1	0.7963	0.7855
CWB1	0.7937	0.7843
KO1	**0.7887**	0.7761
EW4	0.7802	0.7691
EW3	0.7718	0.7587
UUOS1	0.7688	0.7567
OFG1	0.7551	0.7404
MK1	**0.7511**	0.7363
EW1	0.7476	0.7337
PVM1	0.7366	0.727
EW2	0.7296	0.7158
RRHS1	0.7263	0.7128
PP1	0.7023	0.6834
RRHS2	0.5863	0.5729

Table 5.17: MIREX 2010 results in Audio Chord Detection.

in comparison with the previous years. The best "overlap ratio" of 80.2% showed the system **MD1** of Mauch and Dixon. In comparison with MIREX 2009, where the best achieved result was 74.2%, a significant increase of 6% was observed. Needless to mention the fact that the average performance of the submitted systems is significantly higher than a year before. **KO1** and **MK1** systems showed 78.9% and 75.1% overlap ratios respectively. In comparison with the MIREX 2008, where the difference in performance between our "pretrained" and "train-test" systems was about 8%, here we can observe only 3.8%.



(a) Overlap ratio



(b) Weighted average overlap ratio

Figure 5.20: Tukey-Kramer HSD test for MIREX 2010 results.

5.7.4 MIREX 2011

MIREX 2011 contest in audio chord detection is an exact replica of MIREX 2010 in terms of evaluation metrics and datasets.

Participants from different teams are presented in Table 5.18. The results are given in Table 5.19. Tukey-Kramer HSD tests for statistical significance are depicted in Figure 5.21.

Our chord detection systems are marked as **KO1** and **KO2**. As opposed to our submissions to MIREX 2010, where RC features were used as the front-end, HRC features were adopted. The same multi-stream HMM configuration as in MIREX 2010 was utilized. Similarly to the previous year, we submitted two systems. **KO1** system was submitted pretrained, while **KO2**

Team ID	Authors
BUURO1 – BUURO5	I. Balazs, Y. Ueda, Y. Uchiyama, S. Raczynski, N. Ono, S. Sagayama
CB1 – CB3	T. Cho, J. P. Bello
KO1 – KO2	M. Khadkevich, M. Omologo
NM1	Y. Ni, M. Mcvicar
NMSD1 – NMSD3	Y. Ni, M. Mcvicar, R. Santos-Rodriguez, T. De Bie
PVM1	J. Pauwels, M. Varewyck, J.-P. Martens
RHRC1	T. Rocher, P. Hanna, M. Robine, D. Conklin
UUOS1	Y. Ueda, Y. Uchiyama, N. Ono, S. Sagayama
UUROS1	I. Balazs, Y. Ueda, Y. Uchiyama, S. Raczynski, N. Ono, S. Sagayama

Table 5.18: Team legend for MIREX 2011 audio chord detection contest.

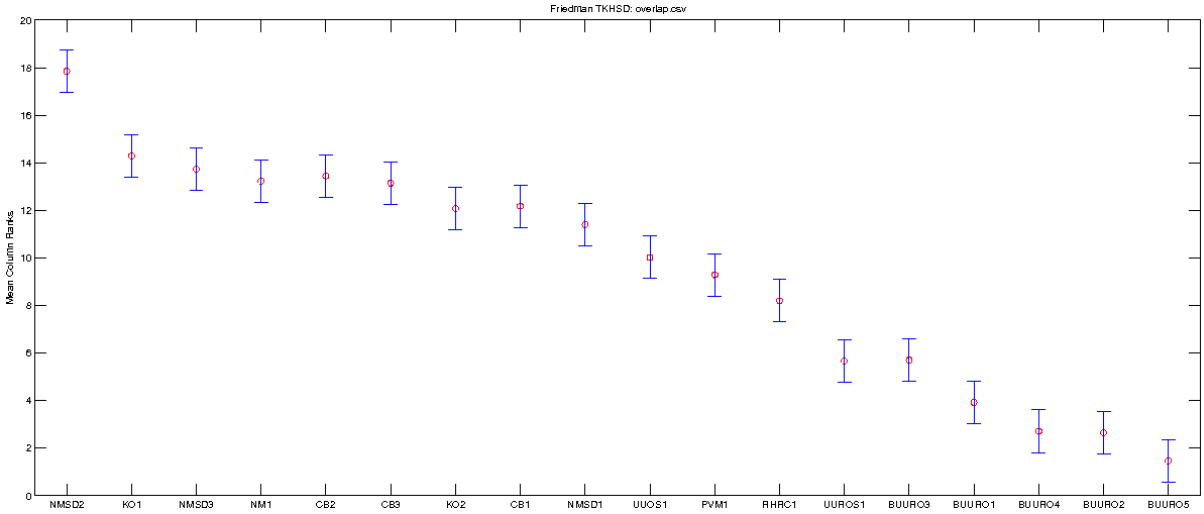
Algorithm	Chord Overlap ratio	Chord weighted average overlap ratio
NMSD2	0.976	0.9736
KO1	**0.8285**	0.8163
NMSD3	0.8277	0.8197
NM1	0.8199	0.8114
CB2	0.8137	0.8
CB3	0.8091	0.7957
KO2	**0.7977**	0.7822
CB1	0.7955	0.7786
NMSD1	0.7938	0.7829
UUOS1	0.7689	0.7564
PVM1	0.7396	0.7296
RHRC1	0.7289	0.7151
UUROS1	0.3429	0.3386
BUURO3	0.3427	0.3385
BUURO1	0.2361	0.2313
BUURO4	0.1898	0.1853
BUURO2	0.1675	0.1616
BUURO5	0.1264	0.1215

Table 5.19: MIREX 2011 results in Audio Chord Detection.

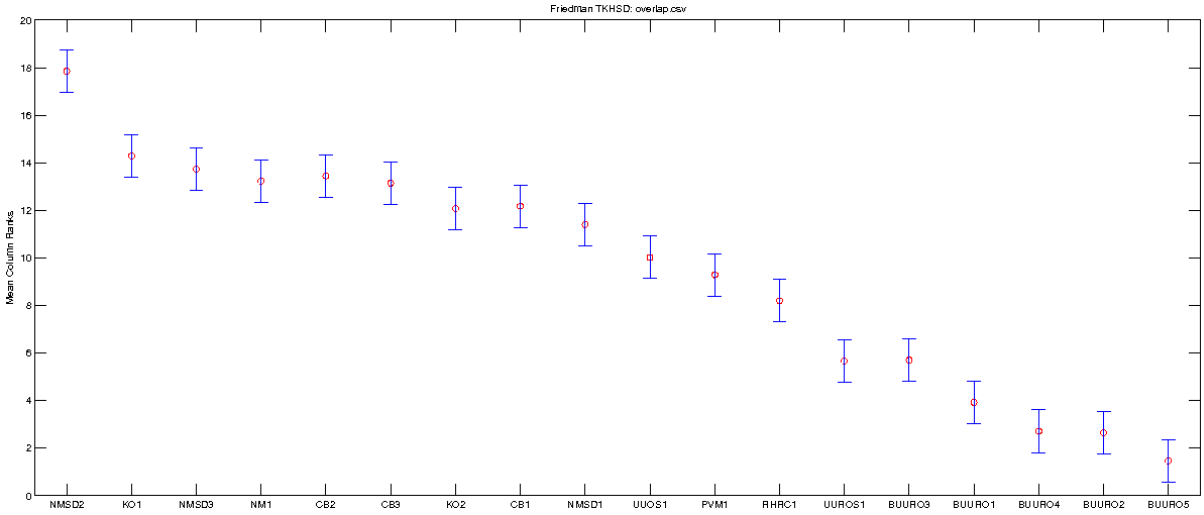
system was submitted for 3-fold cross-validation.

The highest overlap ratio showed system **NMSD2** of Ni et al. Almost perfect chord transcription was demonstrated with the overlap ratio of 97.6%. However, it is probable that the system used ground-truth labels along with some song identification algorithm, assigning ground-truth chord progression to the identified song. This can be guessed from the fact that the majority of the transcribed songs had 100% overlap ratio. For the songs that the system could not identify properly we can observe inconsistencies in the duration of output labels, for example, the output labels for an audio file of 120 seconds could be 170 seconds.

Among the rest of the systems that are based solely on the audio content analysis, **KO1**



(a) Overlap ratio



(b) Weighted average overlap ratio

Figure 5.21: Tukey-Kramer HSD test for MIREX 2011 results.

showed the highest overlap ratio of 82.85%. The next results are pretty much close to this value and do not show statistically significant differences as shown in Figure 5.21. We can observe further improvement of chord recognition system performance in comparison with MIREX 2010.

Chapter 6

Conclusions

This chapter summarizes contributions of the first part of the thesis. Possible directions for future work are outlined.

6.1 Summary of the contributions

In Chapter 2 we reviewed state-of-the-art approaches to automatic chord recognition. Classification into template-matching, statistical, and hybrid approaches was provided. General information on feature vector selection and extraction techniques for automatic chord recognition was given. The importance of mistuning estimation problem was highlighted.

Chapter 3 was concerned with different frond-end configurations. Phase-change based method for mistuning rate estimation was proposed. A new class of chroma features that is based on the PQMF filter bank and Time-Frequency Reassigned spectrogram was introduced. Detailed description of feature vector extraction using the proposed methods was provided. The main contribution of this chapter is the introduction of two novel chroma features.

In Chapter 4 we presented a probabilistic approach to automatic chord recognition and introduced two-level system architecture. Acoustic modeling approach base on multi-stream HMMs was described. Application of standard and factored language models was outlined. Finally, general overview of the proposed chord recognition system was given.

In Chapter 5 we performed a systematic evaluation of different system configurations. We investigated the influence of different parameters on the system performance. The experimental results show that chroma extraction based on PQMF filter bank analysis and subsequent periodicity detection does not outperform the standard approach for the analysis frame length of 182 ms. However, when taking into consideration short-term analysis with frame lengths of 46 ms and 92 ms the proposed approach significantly outperforms the application of standard chroma feature. The TFR technique proved to be effective for producing more accurate chroma features that outperformed the traditional one. A novel approach for harmonic component separation in

the spectral domain that was used for generating HRC chroma feature showed the best performance. Tolerance factor impact on the HRC feature performance was addressed and an optimal choice has been individuated. Another interesting investigation was carried out in the acoustic modeling. The multi-stream HMM structure for chord recognition system, where the two observable layers represent harmonic content of two frequency regions was evaluated and showed better performance in comparison with the single-stream HMM structure. Experimental results showed that assigning different weights to different feature streams influences the recognition rate. We proved the fact that accurate spectral analysis for feature extraction can significantly improve chord recognition accuracy. Large-scale evaluations of chord recognition systems with different parameter configurations pointed out the optimal settings, which imply HRC chroma feature with multi-stream HMM, where the two observable streams correspond to chroma and bass-chroma. A substantial improvement over the baseline system has been obtained with the final result of 81.58% recognition rate. The proposed system showed the highest overlap ratio in MIREX 2011 competition among chord recognition systems, which are based solely on audio content analysis.

6.2 Future work and perspectives

MIREX competitions during the past 4 years show a notable trend to continuous improvement of different submitted chord recognition systems. Starting from 72% of the best overlap ratio in 2008, we can observe an increase of about 10% nowadays.

The most straightforward possible improvement can be brought to the system by including probabilistic modeling of temporal structure. This can be done by introducing an additional hidden layer in HMMs, where hidden states correspond to different beat phases. Additional feature vector stream for modeling observation probabilities of beat events will be introduced. An interesting research could be carried out in the area of possible interaction and mutual dependencies of different hidden layers in HMMs.

Another interesting direction of future work considers further improvement of the feature vectors quality. Careful analysis of higher harmonics can be performed using the proposed TFR technique. Applying higher harmonic subtraction can lead to even better performance, as was shown in [34, 35].

Part II

Beat structure extraction

In this part of the thesis we suggest an approach that performs simultaneous estimation of beats and downbeats. It consists of two hierarchical layers, which include acoustic modeling and beat sequence modeling, and proposes a novel schema to model periodic metrical structure.

Chapter 7

Background

Extracting different types of semantic information from music data has become an emerging area of research in Music Information Retrieval (MIR) community. Tempo estimation and beat/downbeat tracking are amongst the most challenging tasks in MIR community. While processing modern rock and pop songs with rich percussive part and stable rhythm is a nearly solved problem, dealing with non-percussive music with soft note onsets and time-varying tempo, that is characteristic of classical music, is still a challenge.

As opposed to tempo estimation, where only the periodicity of beats is looked for, beat/downbeat tracking implies also producing correct time positions corresponding to rhythmical events. A notion of beats can be defined as time instants, when human being taps his or her foot trying to follow the music. From the musicological viewpoint, downbeat position is defined as the first beat in a bar. Classification of rhythmical events into beats and downbeats brings a portion of useful information about metrical structure, that can be used as high-level feature in many MIR tasks.

There are lots of different approaches for beat/downbeat extraction. Most of them are based on searching for periodicities in some kind of Onset Detection Function (ODF) [82], [83]. The most common periodicity detection methods are based on autocorrelation [84], [85], bank of comb filter resonators [86], or short-time Fourier transform of the ODF [84]. All the methods aim at revealing periodicities in the onset-energy function, from which beat positions and tempo can be derived. The intensities of the estimated periodicity are not constant over time and can be visually represented by means of spectrogram-like representations called rhythmogram [87]. However, estimating beat structure for non-percussive sounds, especially with soft note onsets, becomes a more complex problem due to the noisy ODF. In order to circumvent this, more sophisticated methods that are based on pitch [88] and group delay [89] analysis were proposed.

A lot of attention has been paid to the problem of downbeat tracking. Most approaches are based on some prior knowledge, extracted on previous steps or given to the system as input parameters [90].

Dixon [91] proposed a system that is able to estimate the tempo and the beat locations in expressively performed music. His system can manage either symbolic data such as MIDI, or raw audio. The processing is performed off-line to detect the salient rhythmic events. Then, different hypotheses about the tempo at various metrical levels are generated. Based on these tempo hypotheses, a multiple hypothesis search was applied to find beat locations that fit to the rhythmic events in the best way. In his approach, multiple beat agents compete with each other in the prediction of beat events. His system was tested on a dataset containing songs belonging to different musical styles.

Ellis and Poliner [92], [58] proposed a beat-tracking system that is based on global tempo estimation. The global tempo is extracted using the autocorrelation of ODF function. Then they apply dynamic programming to locate beat positions in the whole song so that beats are placed at the time instants with high ODF values, at the same time keeping spaces between beats that correspond to the global tempo.

Goto [5] described another multiple agent-based beat tracking system that recognizes a hierarchical beat structure. His system is capable of real-time processing. The analysis is performed on several layers: beat, half-bar, and bar. The proposed system can manage audio data with and without drums. Onset times, chord changes, and drum patterns are used to derive hierarchical beat structure. Onset positions are represented by seven-dimensional onset-time vector, where dimensions correspond to the onset times across seven parallel frequency sub-bands. Tempo is estimated using the autocorrelation of the onset sequence. For half-bar and bar detection, bass drum and a snare drum events are detected and matched against drum templates. For non-percussive sounds a measure of chord change probability is used. The underlying ideas are supported by the fact that chord changes occur most commonly on bar positions.

Recently, several HMM-based approaches have been proposed. Peeters in [93] proposed used reverse Viterbi algorithm which decodes hidden states over beat-numbers, while beat-templates are used to derive observation probabilities. Yu Shiu and C.-C. Jay Kuo used periodic HMM structure to extract beat locations [94], based on the tempo information obtained on the previous step.

Chapter 8

Feature extraction

The efficiency of many MIR systems depends highly on the choice of acoustic features. Each task has its distinctive characteristics, and appropriate feature selections plays an important role.

This chapter introduces acoustic features for effective and accurate beat/downbeat positions extraction. Three different features are proposed. The first dimension is represented by Onset Detection Function (ODF) that is based on the impulsive part of the reassigned spectrogram. The second and the third dimensions are introduced to model the dynamics of harmonic changes. In order to model fast and slow changes, Chroma Variation Function (CVF) is calculated for short and large context windows. The choice of CVF as a feature vector component is based on the assumption that most harmonic changes that occur inside a piece of music are located on the bar positions.

8.1 Onset detection function

There are several approaches to compute Onset Detection Function in the literature [82], [83]. Some studies have addressed the usefulness of signal decomposition into several frequency bands and subsequent independent analysis in each band. Goto and Muraoka [95] split the spectrogram into several strips and recognizes onsets by detecting sudden changes in energy. Extracted seven-dimensional onset-time vectors are then processed by a multi-agent system. An example of onset-time vector used in their approach is depicted in Figure 8.1

Scheirer [96] implemented a bank of six elliptic filters. The filtering was performed in time domain. In the next step, tempo is extracted using another bank of comb-filters.

Alonso et al. [82] used decomposition of the analyzed signal into several frequency subbands. Decomposition is performed using a bank of 150th order FIR filters with 80 dB of rejection in the stop band. They also suggest to perform harmonic+noise decomposition of the signal, which aims at separating sinusoidal components from residual noise. In the next step, Musical Stress Profile (MSP), which is an analogue of ODF, is extracted. MSP calculation is

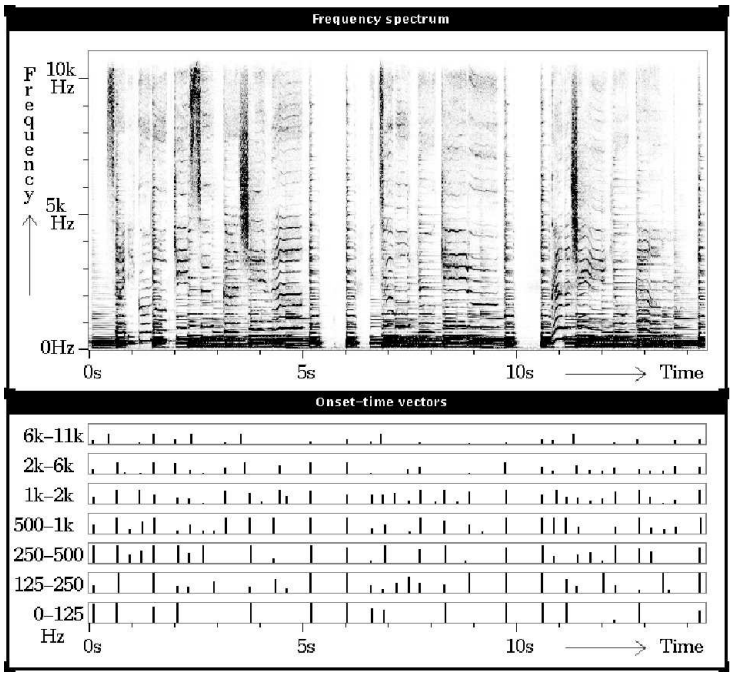


Figure 8.1: Onset-time vector in the approach of Goto and Muraoka.

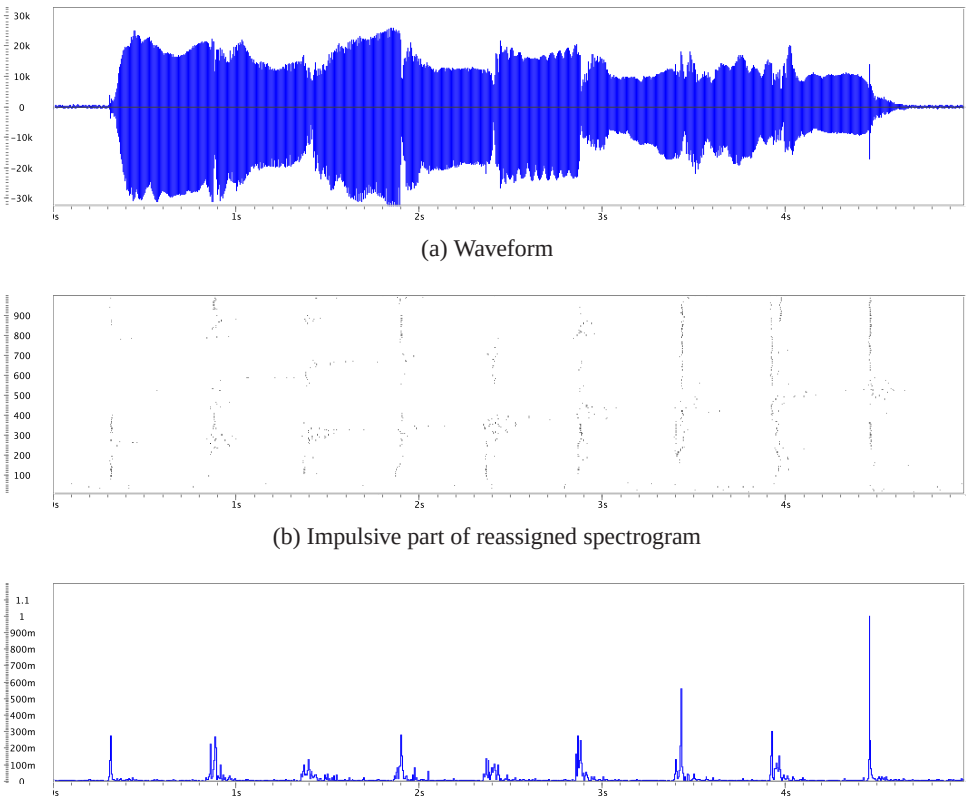
based on the extraction of Spectral Energy Flux (SEF). The choice to use SEF is based on general assumption that the appearance of an onset in an audio stream leads to a variation in the signal's frequency content.

In this work, we propose a novel method to derive onset detection function. It is based on the impulsive part of the reassigned spectrogram.

ODF extraction process starts with transforming audio signal into spectral domain using TFR technique described in Section 3.4. Time-frequency reassigned spectrogram is computed applying impulsive component filtering as shown in Equation (3.24). Having filtered impulsive energy components from the spectrum, onset detection function is obtained by summing all the spectral components in the given frame.

$$ODF(t) = \sum_{k} S_{imp}(t,k) \tag{8.1}$$

where $S_{imp}(t,k)$ is the impulsive spectrogram. Spectral energy sum of the impulsive components acts as the first dimension in the feature vector space.



(a) Waveform

(b) Impulsive part of reassigned spectrogram

(c) Onset detection function

Figure 8.2: Onset detection function of an accending note passage.

8.2 Chroma variation function

Discrimination between beats and downbeats is particularly challenging and often needs a richer feature set, rather than a single ODF. Davies [90] used spectral difference between band-limited beat synchronous analysis frames as a robust downbeat indicator. In this work we propose to use Chroma Variation Function. The main concept here on which we base our ideas is the fact that harmonic (chord) changes occur very frequently on the downbeat positions. CVF reflects the discrepancies between mean chroma vectors of two adjacent segments. This technique was used in [97] and [98], where spectral variation function features were used in for speech recognition and automatic segmentation purposes. It was shown that using variable context lengths along with mean subtraction leads to more robust features. In this paper we adopt a similar approach.

Let $c(k)$ be a chromagram is extracted from the harmonic part of the reassigned spectrogram $S_{harm}(k,n)$ introduced in [59]. Left $c_{l_L}(k)$ and right $c_{r_L}(k)$ contexts of length L correspond to the bins with indexes $[k-L,...,k]$ and $[k,...,k+L]$ respectively.

$$CVF(k) = \frac{1 - \min(M_{left}, M_{right})}{2} \tag{8.2}$$

where:

$$M_{left} = \min_{1 \leq j \leq L} (\rho({c_{l_L}}'(k), {c_{r_j}}'(k))) \tag{8.3}$$

$$M_{right} = \min_{1 \leq j \leq L} (\rho({c_{l_j}}'(k), {c_{r_L}}'(k))) \tag{8.4}$$

In these equations ${c_{l_j}}'(k)$ and ${c_{r_j}}'(k)$ are the left and the right contexts with subtracted mean value over time $m(k)$ of the context that corresponds to the bins with indexes $[k-L, ..., k+L]$.

$${c_{l_j}}'(k) = c_{l_j}(k) - m(k) \tag{8.5}$$

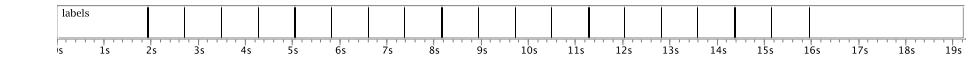
$${c_{r_j}}'(k) = c_{r_j}(k) - m(k) \tag{8.6}$$

when $\rho(c_l, c_r)$ is the normalized inner product between the two context means:

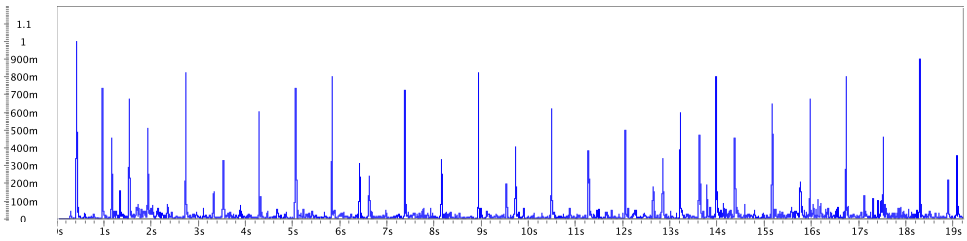
$$\rho(c_l, c_r) = \frac{< \bar{c_l}, \bar{c_r} >}{|\bar{c_l}||\bar{c_r}|} \tag{8.7}$$

The meaning of $CVF(k)$ can be interpreted as a cosine of an angle between the two mean chroma vectors with subtracted $m(k)$ value. In order to identify the highest (i.e., most significant) chroma variations, given the left and the right contexts, minimum values in Equations (8.3) and (8.4) are used. Varying context length L allows one to set up the ability to detect smooth or fast harmonic changes.

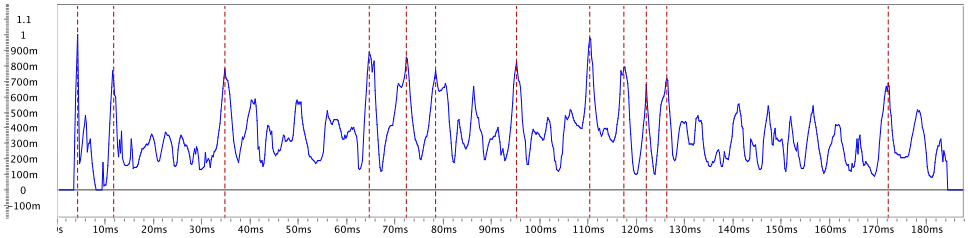
An example of ODF and CVF features extracted from George Michael's "Careless Wisper" are shown in Figure 8.3. Plot 8.3a depicts ground-truth labels for the analyzed excerpt. Thick vertical lines correspond to downbeat positions, while thin lines show beat locations. Onset Detection Function extracted from the excerpt is shown in Figure 8.3b. In the next two plots, Chroma Variation Functions with context lengths of 0.4 and 2 sec are depicted. Vertical dotted lines correspond to the time instants, where there is a local peak in CVF.



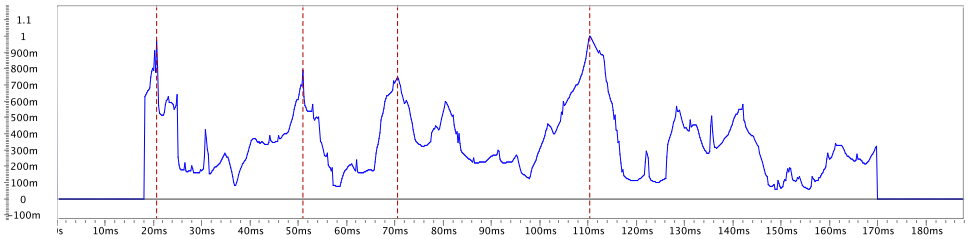
(a) Ground-truth labels



(b) ODF extracted from waveform



(c) CVF with context length of 400 ms.



(d) CVF with context length of 2000 ms.

Figure 8.3: Different feature components extracted from George Michael’s "Careless Wisper".

Chapter 9

System architecture

This chapter describes the proposed statistical approach to automatic extraction of beat sequence from audio. A system architecture that consists of two hierarchical levels, acoustic modeling and beat sequence modeling is introduced. As opposed to deterministic approaches, where beat locations are obtained by periodicity analysis and subsequent beat locating using the tempo information extracted on the previous step, no prior information is needed in the proposed scheme. A specific dictionary and unit alphabet for applying language model approaches are introduced.

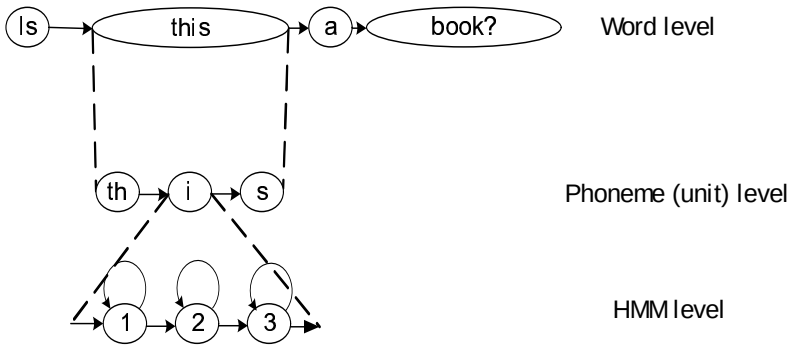
The proposed system is capable of simultaneous extraction of beat and downbeat rhythmic events. A dictionary of beat words is introduced, where different words represent time segments between two adjacent beat events. Similarly to speech recognition, a unit-based transcription of each beat word from the dictionary is provided. The alphabet includes 5 units (beat pre-attack/attack, downbeat pre-attack/attack, no-beat) and beat words are then defined by aggregating units. Each beat word is then characterized by a given duration. In order to model the periodicity of beat events, language modeling block is utilized. Beat word sequence statistics extracted from ground-truth material are used to train N-gram language models.

Section 9.1 introduces acoustic modeling approach adopted here. Section 9.2 is devoted to language modeling techniques. The overview and detailed description of the proposed beat/downbeat extraction system is then presented in Section 9.3.

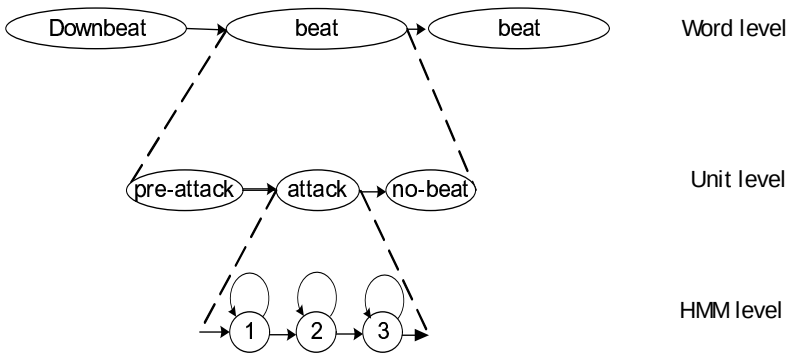
9.1 Acoustic modeling

This section describes the process of building acoustic models for beat/downbeat detection. Two different approaches are introduced. In both approaches an analogy between beat/downbeat detection and speech recognition is drawn, based on the following relationship: phoneme, word, and sentence in speech correspond to unit, beat word, and beat sequence respectively. Figure 9.1 depicts different description levels for a speech sentence and a beat sentence respectively. Two

beat word classes are introduced, which are beats and downbeats. Each beat word from the beat dictionary is characterized by type and duration. The next level of word segmentation is the phoneme-based or unit-based level. At this level, different units that comprise beat words are modeled by a number of hidden states in HMMs.



(a) Speech sentence



(b) Beat sequence

Figure 9.1: Description levels for a speech sentence and a beat sequence.

In the following sections we introduce two different approaches to acoustic modeling. The first approach is based on word-level modeling, while the second approach takes advantages of unit-based acoustic modeling.

9.1.1 Word-based acoustic modeling

The first approach is based on using a dictionary consisting of the two words: beat (BT) and downbeat (DBT). In the training stage audio data is segmented according to the ground-truth labels so that each segment contains time interval between two adjacent beat markers. Two separate left-to-right HMMs that correspond to BT and DBT models are trained using feature vectors extracted from the training material. Each model consists of 3 hidden states. They are supposed to model beat/downbeat attack, sustain and pre-attack phase of the next beat/downbeat. However, no unit-level segmentation information is used in the step of training. All the emission

probabilities are learned from the data using Baum-Weltch algorithm. In the test stage trained HMMs are used by Viterbi decoder to output beat sequences. The block diagram of the described system is depicted in Figure 9.3. HMM training and model connection is schematically represented in Figure 9.2.

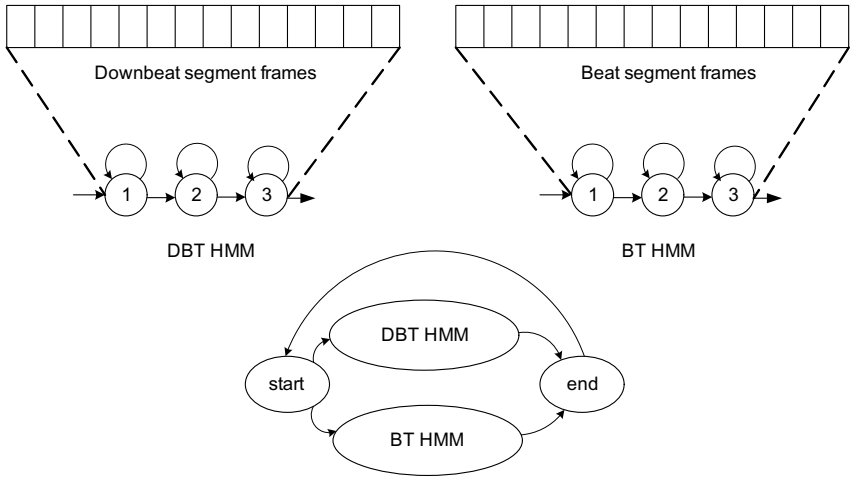


Figure 9.2: Word-level acoustic modeling.

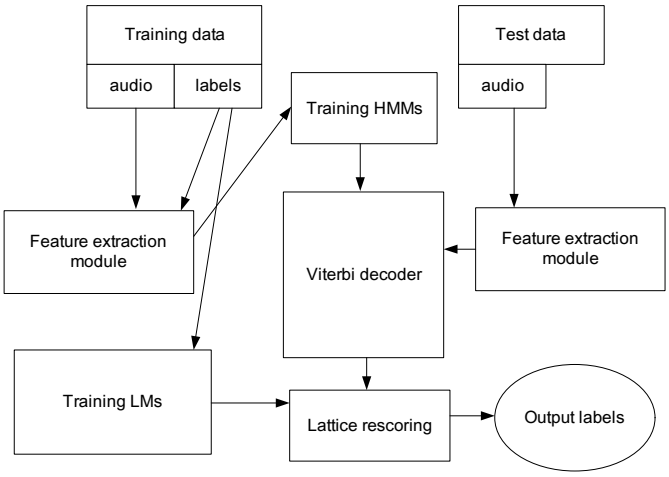
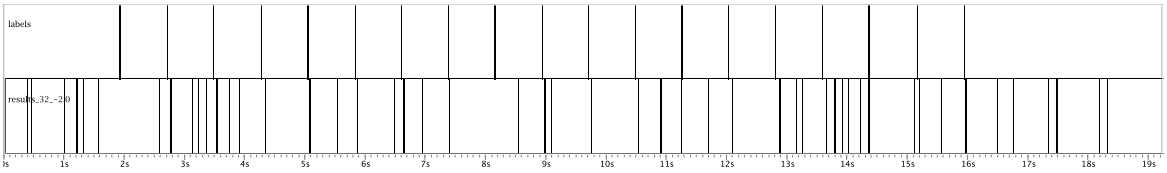


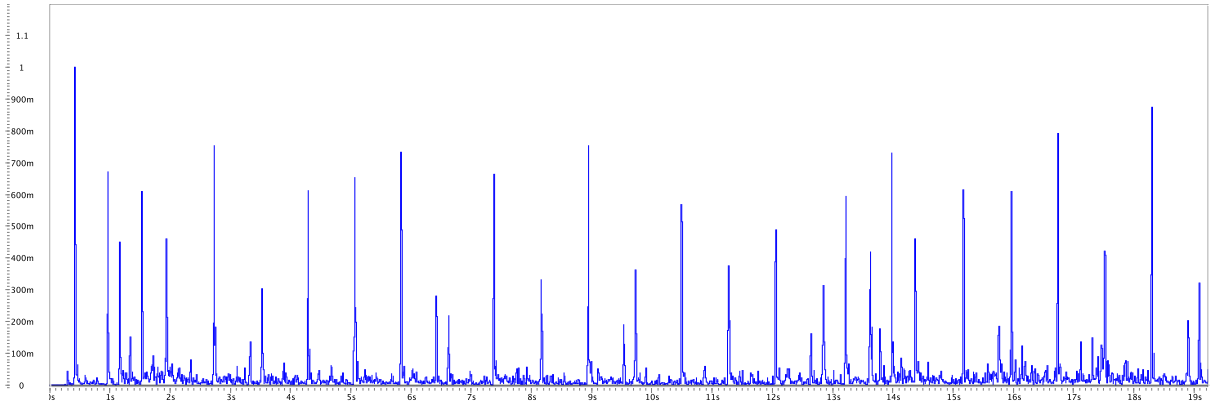
Figure 9.3: Block diagram of beat transcription system.

An output example of the above-described system is shown in Figure 9.4. The first results revealed the fact that the problem of producing periodic output exists and the need for adapting the structure of HMMs is evident. As opposed to the speech recognition task, where word durations can vary significantly and do not influence the overall performance, beat/downbeat detection has some distinctive features. One of the most serious problems one can come across, when trying to use HMM for decoding highly periodical events, is the problem of keeping

periodicity in the output labels. Self-transitions in the states of an HMM allow the model to remain in the same state for quite a long period of time. At the same time, some intervals with numerous note onsets can produce quite dense estimated beat output, as shown in Figure 9.4.



(a) Ground-truth and output labels



(b) Onset detection function

Figure 9.4: An example of the transcription output of George Michael's "Careless Wisper".

9.1.2 Unit-based acoustic modeling

Word-based acoustic modeling that was described in the previous section outlined the problem of periodicity in the output labels. There were some attempts to address this problem in HMM-based approaches. Y. Shiu et al. [94] proposed periodic left-to-right model that produces periodic output. However, a prior information on the tempo is required.

The solution proposed here is to take advantage of unit-based acoustic modeling, to discard all self-transitions in HMMs, and to add an additional beat sequence modeling layer to the system architecture. In this approach, a unit dictionary is constructed, where different units model the following events: beat pre-attack (BTp), beat attack (BTa), downbeat pre-attack (DBTp), downbeat attack (DBTa) and "no beat" (NB). We draw an analogy between a unit in the beat extraction task and a phoneme in speech recognition as was shown in Figure 9.1b.

All the units, apart from NB, are represented by a left-to-right HMM with a number N_{st} of hidden states and no self-transitions. The NB unit has only one emitting state. The number of states N_{st} imposes a duration constraint and corresponds to the necessary number of time frames to output the unit.

Model parameter estimation utilizes training material with ground-truth markers, labeled manually. Extracted feature vectors are segmented according to the ground-truth labels so that each segment contains N_{st} frames corresponding to a specific unit. All the emission probabilities are learned from the training data using Baum-Welch algorithm.

In such a way, different units model different phases of beat/downbeat event, at the same time following the duration constraint. The proposed training schema rules out the possibility to stay in any state for more than one frame. Figure 9.5 depicts an example of acoustic modeling, where $N_{st} = 4$ and $n(i)$ is the number of frames used to train the NB unit in i-th ground-truth beat segment.

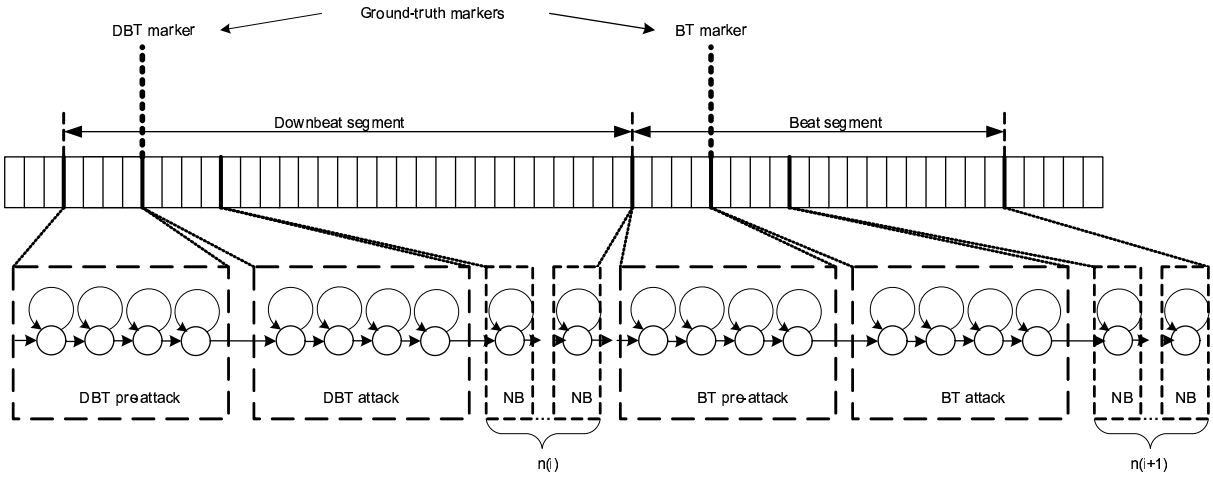


Figure 9.5: Unit-level acoustic modeling.

9.2 Language modeling

Unit-based acoustic modeling approach that was described in the previous section needs high-level language modeling to aggregate units in beat words and to introduce beat duration factor.

Language modeling layer is an essential part in the proposed beat detection system. Its main target is to provide statistical information about beat sequences and beat periodicity. The dictionary for the beat/downbeat tracking task consists of two word classes: beat and downbeat words. Each word from the dictionary is characterized by the duration information.

For each word a unit-level transcription is provided. It consists of a pre-attack unit, followed by an attack unit and a number d_b of NB units that define the duration factor. The first 7 words of the dictionary are provided in Table 9.1.

Having ground-truth annotations for both beats and downbeats, one can collect the statistics on possible beat word sequences. Language model training starts with the extraction of beat sequences from the ground-truth labels. Each beat sequence is composed of beat words defined

Table 9.1: Dictionary for the beat/downbeat tracking task

Word	Unit transcription
beat20	BTp BTa 20NB
downbeat20	DBTp DBTa 20NB
beat21	BTp BTa 21NB
downbeat21	DBTp DBTa 21NB
beat22	BTp BTa 22NB
downbeat22	DBTp DBTa 22NB
beat23	BTp BTa 23NB

as described above. The duration information for each beat word is extracted from the time instants corresponding to the boundaries of the segment.

In order to take into account all possible tempo variations, scaling factor s_f in the range $[0.8 - 1.2]$ with the step of 0.05 are applied. Let us assume a ground-truth beat sequence that consists of only three beat words that are downbeat50, beat50, beat50. The duration d_b of each beat word is equal to 50 frames. After applying scaling procedure with different scaling factors s_f, a number of beat sentences are obtained. Duration $d_f(i)$ of beat words in i-th sentence is defined as $d_f(i) = d_b s_f(i)$. As a consequence, a number of beat sequences is extracted from each ground-truth song. An example of the training material extracted from a short song is given in Table 9.2. Symbols $< s >$ and $< /s >$ denote the beginning and the end of a musical piece respectively. Extracted material is given as an input to train N-gram language models.

Table 9.2: Text extracted from the ground-truth labels

$< s >$ downbeat52 beat52 beat52 beat52 downbeat52 ... beat52 $< /s >$
$< s >$ downbeat54 beat54 beat54 beat54 downbeat54 ... beat53 $< /s >$
$< s >$ downbeat56 beat55 beat56 beat55 downbeat55 ... beat55 $< /s >$
$< s >$ downbeat57 beat57 beat57 beat57 downbeat57 ... beat57 $< /s >$
...
$< s >$ downbeat94 beat94 beat94 beat94 downbeat94 ... beat94 $< /s >$
$< s >$ downbeat96 beat96 beat96 beat96 downbeat95 ... beat95 $< /s >$
$< s >$ downbeat98 beat97 beat98 beat97 downbeat97 ... beat97 $< /s >$

The proposed approach, which includes acoustic and language modeling impose duration constraints and solves the problem of keeping periodicity in the output labels. For example, in order to output beat23 word in the process of decoding it is necessary for the system to start in BTs unit model, remain there for the time corresponding to N_{st} frames, continue in BTa unit model, remain there for another N_{st} frames, and finally switch to 23 successive NB unit models. The absence of self-transitions in HMMs allows for defining duration constraints. Such an explicit duration modeling allows one to have as an output labels with stable duration. The proposed language modeling approach is flexible and N-gram models can be trained on many musical styles. For example, while working with modern pop and rock music, one can

observe stable tempo. However, when dealing with other musical styles such as classic one can observe frequent tempo changes. The main advantage of the proposed approach is the absence of imposed deterministic rules. All the system parameters are estimated using the training data.

9.3 Beat/downbeat detection

The process of beat structure extraction starts with feature vector extraction for a given test song as described in Section 8. Extracted feature vectors are then passed to the decoder. Similarly to the approach of multiple-pass decoding, which has been successfully used in speech recognition [71], the decoding procedure consists of two steps. In the first step, time-and-space efficient bigram language model is applied in the stage of Viterbi decoding, producing a lattice. Lattice nodes denote time instants and lattice arks denote different hypotheses about beat and downbeat events. In the second step, the obtained lattice is rescored applying more sophisticated 4-gram language models on the reduced search space. Finally, the obtained transcription labels are matched against ground-truth. A block-diagram of the system is presented in figure 9.6.

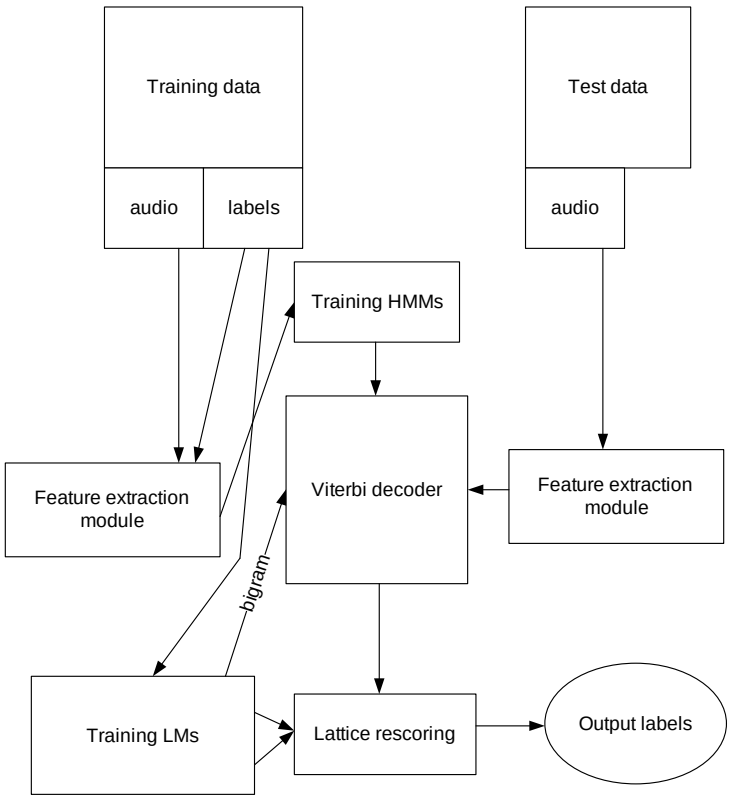


Figure 9.6: Block diagram of the modified beat transcription system.

9.4 Reference system

For the sake of objective evaluation, we compare the proposed beat detection system with a reference system. The reference system we used is the Sonic Annotator software[1] with *Bar and Beat Tracker* QM Vamp plug-in[2]. It will be referred to as "Davies". The plug-in first calculates an Onset Detection Function using the "Complex Domain" method described in [99]. Extracted ODF is processed twice, first to estimate the tempo, and then to estimate beat locations. In the stage of tempo extraction, ODF is segmented into frames of 6 second duration with 75% overlapping. The autocorrelation function of each segment is computed, and then it is passed through a perceptually weighted comb filter bank [99]. Matrix of periodicity observations is produced by the output of the filter bank. In the final step, Viterbi decoder is used to estimate the best path of periodicity through the observations. Given the extracted tempo, beat locations are localized by applying the dynamic programming algorithm described in [85]. Recursive cumulative score function of the ODF is calculated and backtraced on the next step. The cumulative score indicates the likelihood of a beat existing at each sample of the onset detection function input, and the backtrace gives the location of the best previous beat given this point in time. Stark et al. [100] proposed real-time implementation of the above-described beat tracking algorithm. In order to extract bar locations, the third pass processing is performed. The audio signal is down-sampled to 2.8kHz and segmented into beat synchronous frames. Spectral content from each frame is extracted. Spectral difference between adjacent frames is calculated using Kullback-Leibler divergence [90]. Bar locations are then obtained by the analysis of the most consistent spectral change between beats.

[1] http://omras2.org/SonicAnnotator

[2] http://www.vamp-plugins.org

Chapter 10

Experimental results

This chapter is devoted to the evaluation of beat/downbeat detection system introduced in chapter 9. Datasets and evaluation metrics are described in Section 10.1. Experimental results with different feature vector solutions are presented in Section 10.2. Performance of the proposed beat/downbeat extraction system is compared to the reference system introduced in Section 9.4, as well as to other state-of-the-art systems submitted to the MIREX 2011 competition in Audio Beat Tracking.

10.1 Datasets and evaluation metrics

10.1.1 Datasets

Three different datasets were used to evaluate beat extraction system. Each excerpt is manually annotated by expert musicians. The annotations can contain beat annotations only, or accompanying downbeat information.

The first dataset, which consists of 72 modern pop songs, was used for evaluation purposes inside the Quaero project in 2010. The corresponding ground-truth labeling contain two layers, beat and downbeat. Description of the tracks included in the dataset is given in Table A.3.

The second dataset we used is the well-known Hainsworth's dataset. It contains 180 short tracks of different styles, including jazz, folk, and classic. The labels for this dataset contain only beat-level markers, which does not allow us to test downbeat estimation on this dataset.

The third dataset is the ubiquitous Beatles dataset, which contains 172 songs from 12 albums. The corresponding ground-truth annotations were made at Queen Mary University of London. Labels contain both beat and downbeat positions, which makes this dataset suitable for our test purposes.

10.1.2 Evaluation metrics

Following MIREX evaluations, the scoring methods were taken from the beat evaluation toolbox and are described in [101]. Here we provide a short description of each metric used for evaluation.

F-measure is based on the standard **Precision** and **Recall**. Ground-truth annotations are matched against the transcription labels. A precision window of 70 ms is defined. Annotated beat label is considered to be correct if it is located in the interval of $[b_t - pw; b_t + pw]$, where b_t is the ground-truth beat location and pw is the precision window length.

Apart from fixed precision window length of 70 ms, as done under MIREX, we also address an adaptive approach to the calculation of precision window. The precision window is set to 10% of the distance between two successive beat positions in the ground-truth labels. The same evaluation schema is utilized for downbeat evaluation, where precision window is set to 10% of the distance between two successive downbeat positions.

Cemgil - beat accuracy is calculated using a Gaussian error function with 40ms standard deviation as reported in [102].

Goto - binary decision of correct or incorrect tracking based on statistical properties of a beat error sequence.

McKinney's PScore - McKinney's impulse train cross-correlation method as described in [103].

CMLc, CMLt, AMLc, AMLt - continuity-based evaluation methods based on the longest continuously correctly tracked section as introduced in [104].

D, Dg - information based criteria based on analysis of a beat error histogram as described in [101].

10.2 Beat/downbeat extraction

In this section large-scale evaluation of the proposed beat/downbeat detection system is carried out. Different feature vector configuration that are evaluated are given in Table 10.1.

1dimMSP	MSP	-	-
1dim	ODF	-	-
2dim	ODF	CVF 0.4s window	-
3dim	ODF	CVF 0.4s window	CVF 2s window

Table 10.1: Feature vector configurations.

In the first part of the experiments, MSP feature that was described in Section 8.1 is compared with the proposed ODF feature. Experiments with 2dim and 3dim feature vector config-

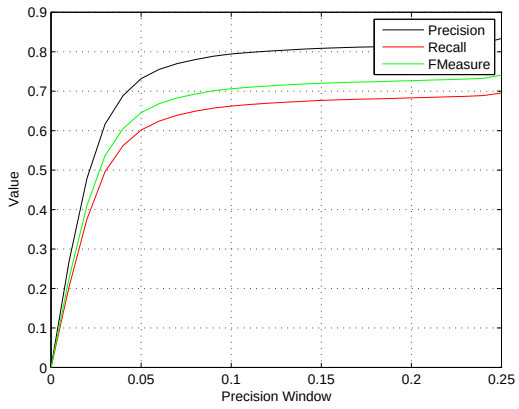
urations are given in Section 10.2.2.

10.2.1 Onset detection function

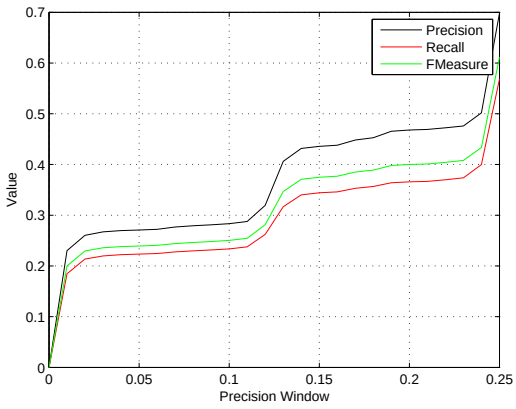
In the first set of experiments we compare the performance of the proposed ODF with the MSP feature described in Section 8.1. The experiments were conducted on the "Quaero" dataset and the results are given in Table10.2.

Algorithm	F-Measure	Cemgil	Goto	McKinney P-score	CMLc	CMLt	AMLc	AMLt	D (bits)	Dg (bits)
ODF	0.7820	68.92	55.56	74.00	40.48	57.25	59.54	86.63	3.03	1.84
MSP	0.7172	66.67	50.00	67.61	37.37	54.16	61.41	85.52	3.00	1.89

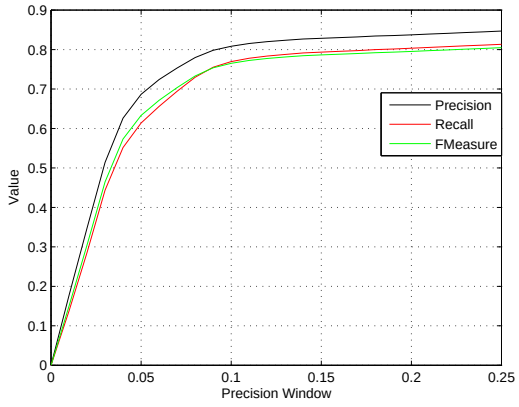
Table 10.2: MIREX-based evaluation results for ODF and MSP features on the Quaero dataset.



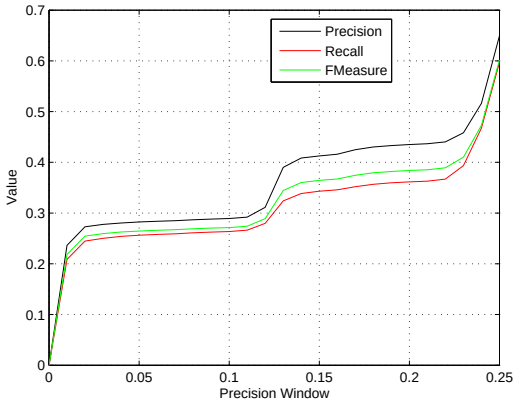
(a) Beat estimation results with MSP feature



(b) Downbeat estimation results with MSP feature



(c) Beat estimation results with ODF feature



(d) Downbeat estimation results with ODF feature

Figure 10.1: Evaluation of MSP and ODF features on the Quaero dataset.

We also estimated precision, recall and F-measure as a function of the precision window length using adaptive approach to the calculation of precision window as was described in Section 10.1.2. The plots are given in Figure 10.1.

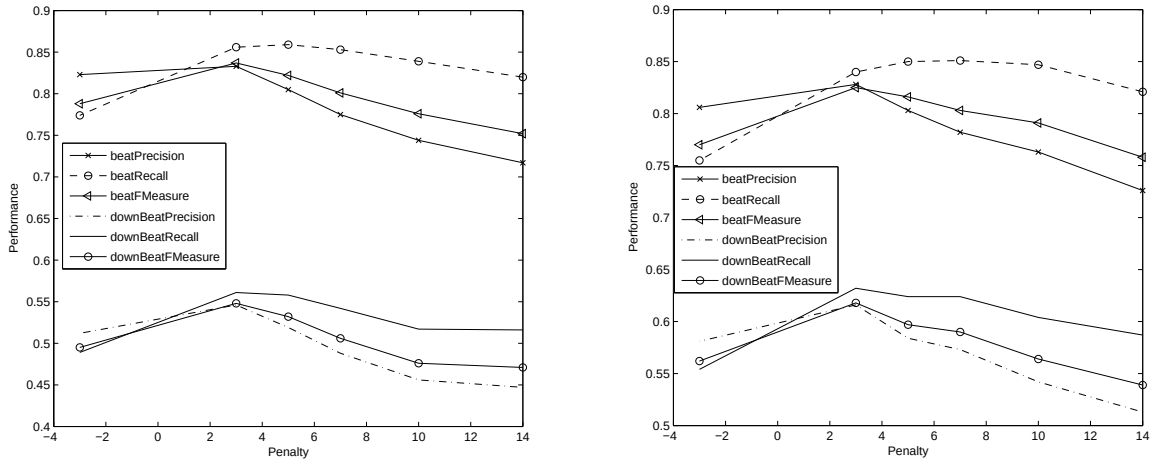
The proposed ODF feature turned out to be quite effective for the accurate estimation of beat positions in comparison with the MSP for the given dataset. However, the performance of downbeat estimation is quite poor. This can be explained by the presence of ODF only in the feature set. ODF or MSP itself cannot model harmonic changes in the signal, which is an essential information for meter estimation.

10.2.2 Chroma variation function

In order to address the problem of poor downbeat estimation performance with a single ODF, CVF vector components with context lengths of 0.4 seconds and 2 seconds were added to the feature set. The results for 2dim and 3dim configurations, as well as for the reference system, are given in Table 10.3.

Algorithm	F-Measure	Cemgil	Goto	McKinney P-score	CMLc	CMLt	AMLc	AMLt	D (bits)	Dg (bits)
2dim	0.8653	81.35	86.11	84.81	74.71	78.68	81.16	85.61	3.05	2.29
3dim	0.8532	80.22	84.72	83.84	72.75	77.10	79.32	84.16	3.00	2.32
DAVIES	0.8723	77.45	80.56	84.15	73.94	76.94	85.11	89.46	3.26	2.24

Table 10.3: MIREX-based evaluation results for 2dim, 3dim and Davies systems on the Quaero dataset.



(a) 2dim feature vector configuration

(b) 3dim feature vector configuration

Figure 10.2: Evaluation results with 2dim and 3dim feature vectors

Figure 10.2 shows the behavior of precision, recall and F-measure for 2dim and 3dim feature

vector configurations as a function of insertion penalty. The optimal value of the insertion penalty in the given system configuration is 3. It shows precision, recall, and F-measure values to be close to each other.

F-measure for downbeat position reached 54% in the case of 2dim feature vector configuration and 10% adaptive precision window. 3dim configuration showed 61.8% F-measure for downbeats. The results for beat estimation indicate that the proposed method with *2dims* feature vector configuration reached the F-measure score very close to the reference system by Davies. However, in evaluation metrics proposed by Cemgil and Goto it achieves better results for the Quaero dataset. Adding CVF feature component with the context length of 0.4 sec to a single ODF shows a significant increase in performance for both beats and downbeats estimation. However, downbeat F-measure is further improved by adding the third feature vector dimension, which is CVF with the context length of 2 sec. Needless to mention a slight decrease in the beats F-measure estimate in comparison with the *2dims* configuration. Nevertheless, these two features do improve the downbeat estimation results for the proposed method.

The next series of experiments were conducted on the "Hainsworth" and "Beatles" datasets. The results are given in Table 10.4 and Table 10.5 respectively. Figures 10.3 and 10.4 depict precision, recall and F-measure as a function of the precision window length using adaptive approach introduced in Section 10.1.2. Summary results from these plots for 10% adaptive precision window are given in Tables 10.7 and 10.6.

Algorithm	F-Measure	Cemgil	Goto	McKinney P-score	CMLc	CMLt	AMLc	AMLt	D (bits)	Dg (bits)
3dims	0.7756	65.05	67.96	77.80	65.24	68.46	74.49	78.19	2.19	1.15
DAVIES	0.7593	61.73	66.85	76.87	62.87	69.46	78.00	87.31	2.31	0.99

Table 10.4: MIREX-based evaluation results for 3dim and Davies systems on the Hainsworth dataset.

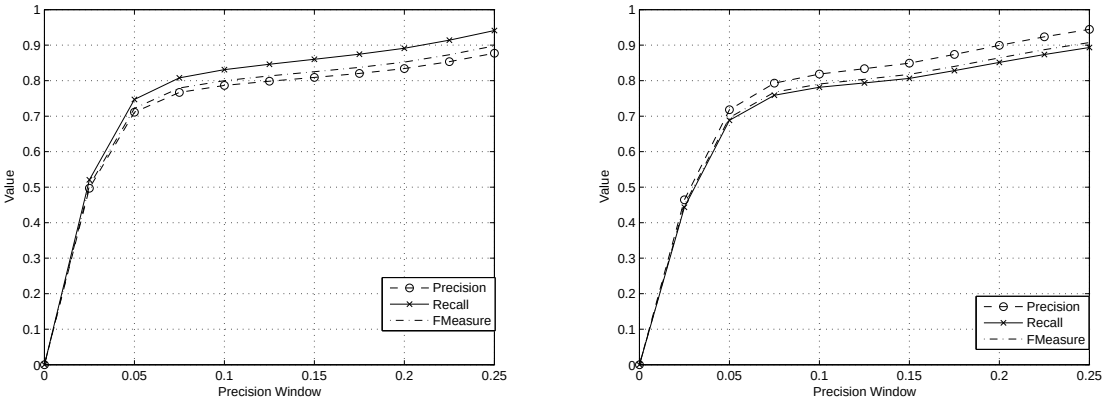
Algorithm	F-Measure	Cemgil	Goto	McKinney P-score	CMLc	CMLt	AMLc	AMLt	D (bits)	Dg (bits)
3dims	82.31	62.88	73.41	83.79	64.61	73.26	72.59	83.81	2.79	1.09
DAVIES	77.03	55.68	61.85	77.22	60.01	68.72	75.48	86.37	3.00	1.18

Table 10.5: Experimental results for 3dim and Davies systems on the "Beatles" dataset.

	Beat F-measure	Downbeat F-measure
3dim	0.8600	0.6082
DAVIES	0.7971	0.6165

Table 10.6: F-measure using 10% adaptive precision window for 3dim and Davies systems on the "Beatles" dataset.

3dim system configuration showed better F-measure for beats than the reference system on the "Hainsworth" collection as shown in Table 10.4, while on the "Beatles" dataset the differ-



(a) Beat estimation results for 3dim system

(b) Beat estimation results for Davies system

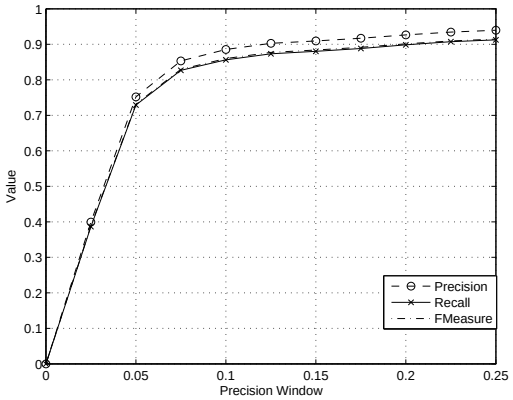
Figure 10.3: Evaluation of 3dim and Davies systems on the Hainsworth dataset.

	Beat F-measure	Downbeat F-measure
3dim	0.8001	-
DAVIES	0.7906	-

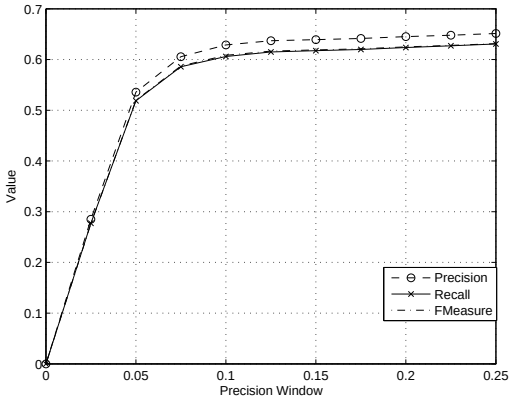
Table 10.7: F-measure using 10% adaptive precision window for 3dim and Davies systems on the "Hainsworth" dataset.

ence in beat F-measures is quite small as shown in Table 10.5. Evaluations with 10% adaptive precision window provided in Tables 10.7 and 10.6 show better F-measure on both datasets for 3dim system. Downbeat F-measures for Davies and 3dim systems on the Beatles dataset are almost equal.

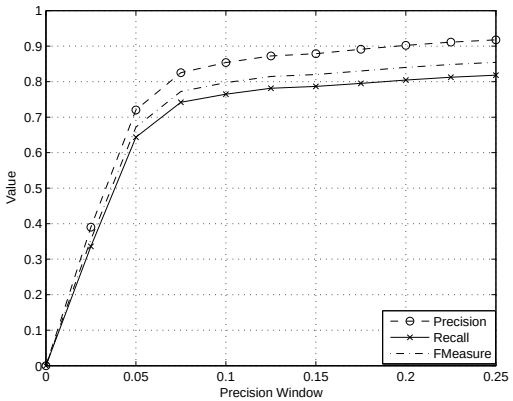
Experimental results showed that the introduced approach to beat/downbeat estimation is effective. The proposed method for ODF extraction proved to be appropriate. It is indicated by the performance of 1dim system for beat extraction. The advantage of additional CVF features is obvious. Experiments showed that the proposed two- and three-dimensional feature vector configurations outperform 1dim system in both, beat and downbeat estimation.



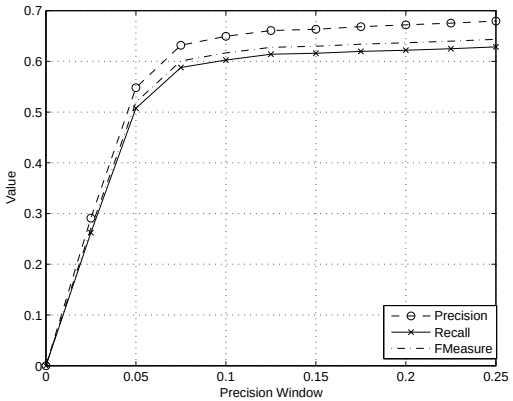
(a) Beat estimation results for 3dim system



(b) Downbeat estimation results for 3dim system



(c) Beat estimation results for Davies system



(d) Downbeat estimation results for Davies system

Figure 10.4: Evaluation of 3dim and Davies systems on the Beatles dataset.

10.3 MIREX 2011

Participation in MIREX 2011 audio beat tracking contest is an excellent opportunity to compare the performance of the developed system with many other systems.

Two different datasets were used. The first dataset, which will be referred to as "MCK", contains 160 30-second excerpts in WAV format used for the Audio Tempo and Beat contests in 2006. Beat locations have been annotated in each excerpt by 40 different listeners. Audio recordings have rather stable tempo that does not significantly change over time. Some examples contain changing meters. The second collection consists of 367 Chopin Mazurkas and will be referred to as "MAZ". In comparison with the "MCK" dataset, most part of the "MAZ" dataset contain tempo changes. MIREX 2011 contest in audio beat tracking does not evaluate downbeat locations estimation.

Participants from different teams are presented in Table 10.8. The results for "MCK" and "MAZ" datasets are given in Tables 10.9 and 10.10 respectively.

Team ID	Authors
FW1	F.-H. F. Wu
GKC2, GKC5	A. Gkiokas, V. Katsouros, G. Carayannis
GP2 – GP5	G. Peeters
KFRO1 – KFRO2	M. Khadkevich, T. Fillon, G. Richard, M. Omologo
SB3 – SB4	S. Böck
SP2	S. Pauws

Table 10.8: Team legend for MIREX 2011 audio beat tracking contest.

Algorithm	F-Measure	Cemgil	Goto	McKinney P-score	CMLc	CMLt	AMLc	AMLt	D (bits)	Dg (bits)
SB3	0.5269	39.92	19.73	57.08	20.83	29.96	37.45	53.64	1.60	0.26
SB4	0.5086	38.55	8.89	54.64	14.18	23.92	24.15	41.64	1.28	0.21
KFRO1	**0.5067**	38.60	18.23	54.75	20.04	27.55	37.27	52.74	1.54	0.26
KFRO2	**0.5045**	38.45	20.00	55.30	24.44	31.33	46.25	57.85	1.70	0.35
GP5	0.5032	37.27	21.18	56.56	23.97	33.69	49.27	66.45	1.81	0.31
GKC2	0.5010	37.83	19.03	55.16	25.81	32.94	51.05	64.23	1.71	0.33
GP4	0.5009	37.00	20.22	56.18	23.26	32.30	48.58	64.89	1.81	0.30
GP3	0.4956	36.65	20.86	56.06	23.36	32.99	47.51	64.89	1.77	0.28
GP2	0.4929	36.38	20.00	55.68	22.85	31.99	46.71	63.47	1.75	0.27
GKC5	0.4854	36.77	15.68	52.83	21.88	29.22	47.55	62.29	1.67	0.31
FW1	0.4784	35.58	6.65	52.36	13.25	22.88	22.64	41.75	1.26	0.18
SP2	0.4353	32.90	8.86	48.16	16.19	23.32	38.96	54.32	1.53	0.26

Table 10.9: MIREX 2011 Results in audio beat tracking contest for MCK dataset.

We have submitted two different systems that are KFRO1 and KFRO2. KFRO1 corresponds to the 2dim feature vector configuration, while KFRO2 correponds to the 3dim one. Model

Algorithm	F-Measure	Cemgil	Goto	McKinney P-score	CMLc	CMLt	AMLc	AMLt	D (bits)	Dg (bits)
FW1	0.6756	57.16	0.31	62.15	7.13	32.92	9.71	40.66	1.46	0.81
SB4	0.5117	42.28	0.00	49.65	4.26	27.85	4.32	28.60	0.54	0.28
GP4	0.4912	37.71	0.31	50.55	3.35	23.89	6.46	34.12	0.49	0.23
GP5	0.4702	36.06	0.00	48.70	3.08	21.24	6.17	31.83	0.43	0.20
GKC2	0.4218	33.50	0.00	41.63	2.21	15.58	5.07	25.98	0.37	0.18
GP2	0.4180	31.56	0.00	44.32	2.52	18.62	4.67	27.13	0.30	0.10
SB3	0.4029	32.70	0.00	40.31	3.27	19.80	3.97	22.79	0.34	0.14
GP3	0.4016	30.35	0.00	42.72	2.29	16.99	4.81	26.54	0.27	0.09
GKC5	0.3731	29.21	0.00	34.85	1.31	7.80	6.21	26.06	0.32	0.15
KFRO2	**0.3504**	28.99	0.31	35.39	2.04	9.45	5.30	20.93	0.31	0.13
SP2	0.3103	24.90	0.00	32.85	1.72	9.77	3.19	16.11	0.22	0.07
KFRO1	**0.2927**	23.15	0.00	29.77	2.16	7.62	4.43	17.49	0.22	0.05

Table 10.10: MIREX 2011 Results in audio beat tracking contest for MAZ dataset.

parameters were estimated using "Quaero" dataset.

Experimental results showed that both systems performed well on the "MCK" dataset, showing F-measure value very close to the top result. However, performance on the "MAZ" dataset turned out to be quite poor. This can be explained by the fact that the systems were trained on a different musical style, which is mostly pop and rock songs. Using classical pieces for model parameter estimation can lead to a better performance on the "MAZ" dataset.

10.4 Tempo estimation based on beat extraction

Tempo is an important piece of information that is coherent with mood or style of a musical excerpt. The most common way of tempo extraction is to analyze ODF for periodicities and estimate the dominant period [82], [105].

This section is devoted to a bottom-up approach to tempo estimation that is based on the above-described beat extraction system. In the first stage, beat/downbeat extraction is performed using 3dim feature configuration, resulting in output labels, from which statistics on the beat lengths is extracted. Since the output transcription may contain segments with different tempos, K-means clustering is performed and the center of the cluster that has the highest number of beats is used to derive the tempo. The approach of Alonso et al. [82] was chosen as a reference system. It will be refereed to as "Alonso".

The test set used for tempo extraction evaluations consists of 961 song excerpts of different musical genres. Each excerpt contains 15-25 seconds of audio of a relatively constant tempo. Part of the collection was used for evaluation purposes in the ISMIR 2004 Tempo Induction Contest. Musical style and tempo are the ground-truth information. The only metric used for evaluation is the percentage of songs with correctly identified tempo. Tempo is treated as correctly identified if estimated value lies within 5% interval of the ground-truth tempo.

The results are presented in Table 10.11.

Genre	Alonso	Bottom-up
1.-Classic	91.1 %	55.2 %
2.-Jazz	96.6 %	83.0 %
3.-Latin music	91.3 %	86.1 %
4.-Pop	97.9 %	92.7 %
5.-Rock	95.6 %	82.4 %
6.-Reggae	100.0 %	100.0 %
7.-Soul	100.0 %	87.5 %
8.-Rap	100.0 %	100.0 %
9.-Techno	98.2 %	87.5 %
0.-Others	96.9 %	83.7 %
1.-Greek	77.9 %	61.4 %
Total	92.50%	76.60%

Table 10.11: Tempo detection results.

The experimental results showed that the proposed bottom-up approach did not succeed and the difference in performance in comparison with the reference system is significant. Tempo estimation based on periodicity analysis of MSP turned out to be more effective. One of the possible reasons for the observed difference in performances could be the fact that parameters of the proposed tempo extraction were estimated on the training material taken from the Quaero corpus, which consists mostly of pop music. That could be the reason for very low performance in classic or Greek part of the test data. However, the proposed bottom-up approach to tempo estimation showed promising performance of 76.60%, which could be improved by training the system on a larger set of songs from different genres.

Chapter 11

Conclusions

In this chapter we summarize the main contributions of the second part of the thesis. Possible future work is discussed.

11.1 Summary of the contributions

Chapter 7 was concerned with existing approaches to beat estimation. Different state-of-the-art systems were reviewed. In Chapter 8 acoustic feature set for effective and accurate beat/downbeat extraction was proposed. A novel approach to the calculation of Onset Detection Function was introduced. It is based on the impulsive part of the reassigned spectrogram. In order to model dynamics of harmonic changes, the usage of CVF feature that is based on the harmonic part of the reassigned spectrum was suggested.

Chapter 9 was devoted to the description of the proposed beat/downbeat extraction system. Two-layered system architecture that comprises acoustic modeling and beat sequence modeling was introduced. Similarities and differences between speech recognition and beat/downbeat extraction are outlined. A specific dictionary and unit alphabet for beat/downbeat extraction was introduced.

Chapter 10 regards the experimental part. Different feature vector configurations were compared with each other, as well as with a reference system using three different datasets. Experimental results showed that the proposed probabilistic approach to simultaneous estimation of beat/downbeat positions from audio is effective. The introduced explicit modeling of beat segment duration in the beat sequence modeling layer proved to be effective for solving the output labels periodicity problem. Participation in MIREX 2011 Audio Beat Tracking context proved the effectiveness of the proposed approach, showing performance very close to the top result on the "MCK" dataset.

A bottom-up approach to tempo estimation that is based on the described beat extraction system was introduced. Evaluations on a large dataset containing music belonging to different

genres showed performance that is significantly lower than that obtained with a reference system. However, these first results could be improved by training the system on a larger dataset.

11.2 Future work and perspectives

One of the possible improvements to the beat/downbeat estimation performance could be considering genre-specific training material. The results of MIREX 2011 showed that our systems did not perform well on the "MAZ" dataset. Training genre-specific models and introducing genre classification block in the system architecture can lead to interesting results.

Another interesting research direction could be in feature selection and adaptation. ODF feature extraction for instruments that are characterizes by soft note onsets can be reprised.

Further improvement could also be gained by incorporating tempo estimation into the model and utilizing high-level features to enhanced downbeat estimation. Another interesting investigation can be conducted in the area of application of multi-stream HMMs, as was shown in the first part of the thesis. Splitting feature vector into a number of separate streams and assigning different stream weights could be effective for beat/downbeat estimation.

Bibliography

[1] A. Wang, "An industrial strength audio search algorithm.," in *ISMIR*, 2003.

[2] N. Orio, "Music retrieval: A tutorial and review," *Foundations and Trends in Information Retrieval*, vol. 1, no. 1, pp. 1–90, 2006.

[3] A. Lacoste and D. Eck, "A supervised classification algorithm for note onset detection," *EURASIP Journal on Advances in Signal Processing*, pp. 1–14, 2007.

[4] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler, "A tutorial on onset detection in music signals," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 5, pp. 1035–1047, 2005.

[5] M. Goto, "An audio-based real-time beat tracking system for music with or without drum-sounds," *Journal of New Music Research*, vol. 30, no. 2, pp. 159–171, 2001.

[6] M. A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney, "Content-Based Music Information Retrieval: Current Directions and Future Challenges," *Proceedings of the IEEE*, vol. 96, pp. 668–696, Mar. 2008.

[7] J. Weil, T. Sikora, J.-L. Durrieu, and G. Richard, "Automatic generation of lead sheets from polyphonic music signals.," in *ISMIR* (K. Hirata, G. Tzanetakis, and K. Yoshii, eds.), pp. 603–608, International Society for Music Information Retrieval, 2009.

[8] I. Simon, D. Morris, and S. Basu, "Mysong: automatic accompaniment generation for vocal melodies.," in *CHI* (M. Czerwinski, A. M. Lund, and D. S. Tan, eds.), pp. 725–734, ACM, 2008.

[9] S. Fukayama, K. Nakatsuma, S. Sako, Y. Yonebayashi, T. H. Kim, S. W. Qin, T. Nakano, T. Nishimoto, and S. Sagayama, "Orpheus: Automatic composition system considering prosody of japanese lyrics.," in *ICEC* (S. Natkin and J. Dupire, eds.), vol. 5709 of *Lecture Notes in Computer Science*, pp. 309–310, Springer, 2009.

[10] J. S. Downie, K. West, A. F. Ehmann, and E. Vincent, "The 2005 music information retrieval evaluation exchange (mirex 2005): Preliminary overview.," in *ISMIR*, pp. 320–323, 2005.

[11] J. S. Downie, J. Futrelle, and D. K. Tcheng, "The international music information retrieval systems evaluation laboratory: Governance, access and security.," in *ISMIR*, 2004.

[12] K. West, A. Kumar, A. Shirk, G. Zhu, J. S. Downie, A. Ehmann, and M. Bay, "The networked environment for music analysis (nema)," in *Proc. 6th World Congress Services (SERVICES-1)*, pp. 314–317, 2010.

[13] J. S. Downie, "The music information retrieval evaluation exchange (2005-2007): A window into music information retrieval research," *Acoustical Science and Technology*, vol. 29, no. 4, pp. 247–255, 2010.

[14] 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 *ICME*, pp. 1505–1508, 2008.

[15] K. Lee, "Identifying cover songs from audio using harmonic representation," in *MIREX task on Audio Cover Song Identification*, 2006.

[16] H. Papadopoulos and G. Peeters, "Local key estimation based on harmonic and metric structures," in *Proceedings of DAFX*, (Como, Italy), 2009.

[17] L. Oudre, Y. Grenier, and C. Févotte, "Template-based chord recognition : Influence of the chord types," in *ISMIR*, 2009.

[18] C. Harte and M. Sandler, "Automatic chord identification using a quantized chromagram," in *Proceedings of the Audio Engineering Society*, (Spain), 2005.

[19] A. Sheh and D. P. Ellis, "Chord segmentation and recognition using em-trained hidden markov models," in *Proc. 4th International Conference on Music Information Retrieval*, 2003.

[20] 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*, vol. 16, february 2008.

[21] H. Papadopoulos and G. Peeters, "Simultaneous estimation of chord progression and downbeats from an audio file," in *Proc. ICASSP*, 2008.

[22] T. Yoshioka, T.Kitahara, K. Komatani, T. Ogata, and H. Okuno, "Automatic chord transcription with concurrent recognition of chord symbols and boundaries," in *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR)*, (Barcelona), 2004.

[23] 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 *ISMIR* (J. P. Bello, E. Chew, and D. Turnbull, eds.), pp. 39–44, 2008.

[24] D. P. W. Ellis, "The 2009 labrosa pretrained audio chord recognition system," in *MIREX Annual Music Information Rerieval eXchange. Avalable at http://www.music-ir.org/mirex/2009/results/abs/DE.pdf*, 2009.

[25] M. Khadkevich and M. Omologo, "Use of hidden markov models and factored language models for automatic chord recognition," in *Proceedings of the 2009 ISMIR Conference*, (Kobe, Japan), 2009.

[26] Y. Ueda, Y. Uchiyama, T. Nishimoto, N. Ono, and S. Sagayama, "Hmm-based approach for automatic chord detection using refined acoustic features," in *Proc. ICASSP*, 2010.

[27] M. Mauch and S. Dixon, "Simultaneous estimation of chords and musical context from audio," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 6, pp. 1280–1289, 2010.

[28] D. P. W. Ellis and A. Weller, "The 2010 labrosa chord recognition system," in *MIREX Annual Music Information Rerieval eXchange. Available at* `http://www.ee.columbia.edu/~dpwe/pubs/Ellis10-chords`, 2010.

[29] T. Fujishima, "Realtime chord recognition of musical sound: A system using common lisp music," in *Proceedings of the International Computer Music Conference*, (Beijing), 1999.

[30] M. Mauch and S. Dixon, "A discrete mixture model for chord labelling," in *Proceedings of the 2008 ISMIR Conference*, (Philadelphia), 2008.

[31] G. Peeters, "Musical key estimation of audio signal based on hmm modeling of chroma vectors," in *Proceedings of DAFX*, (McGill, Montreal, Canada), 2006.

[32] G. Peeters, "Chroma-based estimation of musical key from audio-signal analysis," in *Proceedings of the 2006 ISMIR Conference*, (Victoria, Canada), 2006.

[33] M. Müller, *Information Retrieval for Music and Motion*. Springer Verlag, 2007.

[34] M. Mauch and S. Dixon, "Approximate note transcription for the improved identification of difficult chords," in *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR 2010)*, 2010.

[35] M. Varewyck, J. Pauwels, and J.-P. Martens, "A novel chroma representation of polyphonic music based on multiple pitch tracking techniques," in *MM '08: Proceeding of the 16th ACM international conference on Multimedia*, (New York, NY, USA), pp. 667–670, ACM, 2008.

[36] E. Gómez and P. Herrera, "Automatic extraction of tonal metadata from polyphonic audio recordings," in *AES*, 2004.

[37] J. P. Bello and J. Pickens, "A robust mid-level representation for harmonic content in music signal," in *Proceedings of the 2005 ISMIR Conference*, (London), pp. 304–311, 2005.

[38] H. Papadopoulos and G. Peeters, "Joint estimation of chords and downbeats from an audio signal," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 1, pp. 138–152, 2011.

[39] H. Papadopoulos and G. Peeters, "Large-scale study of chord estimation algorithms based on chroma representation and hmm," in *Content-Based Multimedia Indexing, 2007. CBMI '07. International Workshop on*, pp. 53–60, june 2007.

[40] E. Gómez, "Tonal description of polyphonic audio for music content processing.," *INFORMS Journal on Computing*, vol. 18, no. 3, pp. 294–304, 2006.

[41] L. Oudre, Y. Grenier, and C. Févotte, "Chord recognition using measures of fit, chord templates and filtering methods," in *Proc. IEEE Workshop Applications of Signal Processing to Audio and Acoustics WASPAA '09*, pp. 9–12, 2009.

[42] L. Rabiner, "A tutorial on hidden markov models and selected applications inspeech recognition," *Proceedings of the IEEE*, vol. 77, pp. 257–286, 1989.

[43] H. Papadopoulos, *Joint estimation of musical content information from an audio signal*. PhD thesis, IRCAM, 2010.

[44] M. Mauch, K. Noland, and S. Dixon, "Using musical structure to enhance automatic chord transcription," in *Proceedings of the 10th International Conference on Music Information Retrieval, Kobe, Japan*, pp. 231–236, 2009.

[45] Y. Ni, M. Mcvicar, R. Santos-Rodriguez, and T. D. Bie, "An end-to-end machine learning system for harmonic analysis of music," *CoRR*, vol. abs/1107.4969, 2011. informal publication.

[46] M. Mauch, *Automatic Chord Transcription from Audio Using Computational Models of Musical Context*. PhD thesis, Queen Mary University of London, 2010.

[47] R. Scholz, E. Vincent, and F. Bimbot, "Robust modeling of musical chord sequences using probabilistic n-grams," in *Proc. ICASSP*, 2009.

[48] E. Unal, P. Georgiou, S. Narayanan, and E. Chew, "Statistical modeling and retrieval of polyphonic music," in *Proc. IEEE MMSP*, 2007.

[49] J. L. Flanagan and R. M. Golden, "Phase vocoder," *Bell Syst. Tech.J.*, vol. 45, pp. 1493–1509, november 1966.

[50] M. Müller and S. Ewert, "Chroma Toolbox: MATLAB implementations for extracting variants of chroma-based audio features," in *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR)*, (Miami, USA), 2011.

[51] T. Q. Nguyen and M. Eee, "Near-perfect-reconstruction pseudo-qmf," *IEEE Trans. Signal Processing*, vol. 42, pp. 65–76, 1994.

[52] W. Hess in *Pitch Determination of Speech-Signals: Algorithms and Devices.*, (Springer-Verlag, Berlin.), 1983.

[53] Y. Medan, E. Yair, and D. Chazon, "Super resolution pitch determination of speech signals," vol. 39, pp. 40–48, January 1991.

[54] S. Chung and V. R. Algazi, "Improved pitch detection algorithm for noisy speech," in *Proceedings ICASSP*, pp. 407–410, 1985.

[55] R. D. Mori and M. Omologo, "Normalized correlation features for speech analysis and pitch extraction," in *Visual Representation of Speech Signals, chapter 31*, John Wiley & Sons Ud., 1993.

[56] K. Kodera, R. Gendrin, and C. Villedary, "Analysis of time-varying signals with small bt values," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 26, pp. 64 – 76, feb. 1978.

[57] T. Abe and M. Honda, "Sinusoidal model based on instantaneous frequency attractors," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 14, no. 4, pp. 1292–1300, 2006.

[58] D. P. W. Ellis and G. E. Poliner, "Identifying 'cover songs' with chroma features and dynamic programming beat tracking," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, vol. 4, April 2007.

[59] K. R. Fitz and S. A. Fulop, "A unified theory of time-frequency reassignment," *CoRR*, vol. abs/0903.3080, 2009.

[60] F. Auger and P. Flandrin, "Improving the readability of time-frequency and time- scale representations by the reassignment method," *IEEE Trans. Speech Audio Process*, vol. 5, no. 43, pp. 1068–1089, 1995.

[61] D. J. Nelson, "Instantaneous higher order phase derivatives," *Digital Signal Processing*, vol. 12, no. 2-3, pp. 416 – 428, 2002.

[62] S. A. Fulop and K. Fitz, "Separation of components from impulses in reassigned spectrograms," in *J. Acoust. Soc. Am. 121*, pp. 1510–1518, 2007.

[63] C. Harte, M. Sandler, and M. Gasser, "Detecting harmonic change in musical audio," in *AMCMM*, pp. 21–26, 2006.

[64] C. Harte and M. Sandler, "Automatic chord recognition using quantised chroma and harmonic change segmentation," in *MIREX Annual Music Information Rerieval eXchange. Available at* `http://www.music-ir.org/mirex/abstracts/2009/harte\_mirex09.pdf`, 2009.

[65] D. Mitrovic, M. Zeppelzauer, and C. Breiteneder, "Features for content-based audio retrieval.," *Advances in Computers*, vol. 78, pp. 71–150, 2010.

[66] A. A. Garcia and R. J. Mammone, "Channel-robust speaker identification using modified-mean cepstral mean normalization with frequency warping," in *Proc. Conf. IEEE Int Acoustics, Speech, and Signal Processing*, vol. 1, pp. 325–328, 1999.

[67] M. Khadkevich and M. Omologo, "Phase-change based tuning for automatic chord recognition," in *Proceedings of DAFX*, (Como, Italy), 2009.

[68] T. Cho, R. J. Weiss, and J. P. Bello, "Exploring Common Variations in State of the Art Chord Recognition Systems," in *Proc. Sound and Music Computing Conference (SMC)*, (Barcelona, Spain), pp. 1–8, jul 2010.

[69] J. Goodman, "A bit of progress in language modeling," in *Computer, Speech and Language*, 2001.

[70] F. Jelinek, "Statistical methods for speech recognition," in *MIT Press*, 1997.

[71] D. Jurafsky and J. H. Martin, eds., *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, 2000.

[72] J. Bilmes and K. Kirchoff, "Factored language models and generalized parallel backoff," in *HLT-NAACL*, 2003.

[73] K. Kirchhoff, D. Vergyri, K. Duh, J. Bilmes, and A. Stolcke, "Morphology-based language modeling for arabic speech recognition," in *Computer, Speech and Language*, 2006.

[74] J. Reed, Y. Ueda, S. Siniscalchi, Y. Uchiyama, S. Sagayama, and C.-H. Lee, "Minimum classification error training to improve isolated chord recognition," in *ISMIR*, pp. 609–614, 2009.

[75] S. Dixon, "Onset detection revisited," in *Proceedings of DAFX*, (McGill, Montreal, Canada), 2006.

[76] A. Stolcke, "Srilm. an extensible language modeling toolkit," in *Proc. Intl. Conf. on Spoken Language Processing*, 2002.

[77] H. Abdi and L. J. Williams, *Encyclopedia of Research Design.*, ch. Honestly significant difference (HSD) test, pp. 583–585. Thousand Oaks, 2010.

[78] C. Joder, S. Essid, and G. Richard, "A comparative study of tonal acoustic features for a symbolic level music-to-score alignment," *ICASSP*, 2010.

[79] C. Harte and M. Sandler, "Symbolic representation of musical chords: A proposed syntax for text annotations," in *Proceedings of the 2005 ISMIR Conference*, 2005.

[80] M. S. Puckette and J. C. Brown, "Accuracy of frequency estimate using the phase vocoder," *IEEE Trans. Speech Audio Process*, vol. 6, pp. 166–176, march 1998.

[81] A. Weller, D. Ellis, and T. Jebara, "Structured prediction models for chord transcription of music audio," in *Proc. Int. Conf. Machine Learning and Applications ICMLA '09*, pp. 590–595, 2009.

[82] M. Alonso, G. Richard, and B. David, "Accurate tempo estimation based on harmonic + noise decomposition," *EURASIP J. Appl. Signal Process.*, vol. 2007, pp. 161–175, January 2007.

[83] P. Grosche and M. Müller, "A mid-level representation for capturing dominant tempo and pulse information in music recordings," in *Proceedings of the 2009 ISMIR Conference*, (Kobe, Japan), 2009.

[84] G. Peeters, "Template-based estimation of time-varying tempo," *EURASIP J. Appl. Signal Process.*, vol. 2007, no. 1, pp. 158–158, 2007.

[85] D. P. W. Ellis, "Beat tracking by dynamic programming," *Journal of New Music Research*, vol. 36, no. 1, pp. 51–60, 2007.

[86] A. P. Klapuri, A. J. Eronen, and J. T. Astola, "Analysis of the meter of acoustic musical signals," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 342–355, 2006.

[87] K. Jensen, J. Xu, and M. Zachariasen, "Rhythm-based segmentation of popular chinese music," in *In ISMIR*, 2005.

[88] M. Mattavelli, G. Zoia, and R. Zhou, "Music onset detection based on resonator time frequency image," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1685–1695, 2008.

[89] A. Holzapfel and Y. Stylianou, "Beat tracking using group delay based onset detection," in *ISMIR*, pp. 653–658, 2008.

[90] M. E. P. Davies and M. D. Plumbley, "A spectral difference approach to downbeat extraction in musical audio," in *Proc. EUSIPCO*, 2006.

[91] S. E. Dixon, "Automatic extraction of tempo and beat from expressive performances," vol. 30, no. 1, pp. 39–58, 2001.

[92] D. P. W. Ellis, "Beat tracking with dynamic programming," in *MIREX Annual Music Information Rerieval eXchange. Available at* `http://www.music-ir.org/evaluation/MIREX/2006abstracts/TE_BT_ellis.pdf`, 2006.

[93] G. Peeters, "Beat-tracking using a probabilistic framework and linear discriminant analysis," in *Proc. DAFX*, 2009.

[94] Y. Shiu and C.-C. J. Kuo, "A hidden markov model approach to musical beat tracking," in *Proc. ICASSP*, 2008.

[95] M. Goto and Y. Muraoka, "Beat tracking based on multiple-agent architecture a real-time beat tracking system for audio signals," in *In Proc. Second International Conference on Multiagent Systems*, pp. 103–110, 1996.

[96] E. Scheirer, "Tempo and beat analysis of acoustic musical signals," *The Journal of the Acoustical Society of America*, vol. 103, no. 1, pp. 588–601, 1998.

[97] F. Brugnara, R. D. Mori, D. Giuliani, and M. Omologo, "Improved connected digit recognition using spectral variation functions," in *Proc. ICSLP*, 1992.

[98] F. Brugnara, D. Falavigna, and M. Omologo, "Automatic segmentation and labeling of speech based on hidden markov models," *Speech Communication*, vol. 12, no. 4, pp. 357–370, 1993.

[99] M. E. P. Davies and M. D. Plumbley, "Context-dependent beat tracking of musical audio.," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 15, no. 3, pp. 1009–1020, 2007.

[100] A. M. Stark, M. E. P. Davies, and M. D. Plumbley, "Real-time beat-synchronous analysis of musical audio," in *Proceedings of DAFX*, (Como, Italy), 2009.

[101] M. E. P. Davies, N. Degara, and M. D. Plumbley, "Evaluation methods for musical audio beat tracking algorithms," tech. rep., Queen Mary University of London, Centre for Digital Music, 2009.

[102] A. T. Cemgil, H. J. Kappen, P. Desain, and H. Honing, "On tempo tracking: Tempogram representation and Kalman filtering," *Journal of New Music Research*, vol. 28, no. 4, pp. 259–273, 2001.

[103] M. F. McKinney, D. Moelants, M. E. P. Davies, and A. Klapuri, "Evaluation of audio beat tracking and music tempo extraction algorithms," *Journal of New Music Research*, vol. 36, no. 1, pp. 1–16, 2007.

[104] S. Hainsworth, *Techniques for the Automated Analysis of Musical Audio.* PhD thesis, University of Cambridge, 2004.

[105] Peeters, "Tempo detection and beat marking for perceptual tempo induction," in *MIREX Annual Music Information Rerieval eXchange. Avalable at http://www.music-ir.org/evaluation/mirex-results/articles/tempo/peeters.pdf*, 2005.

Appendix A

Datasets

Table A.1: Beatles dataset.

CD	Album name
1	Please Please Me
2	With the Beatles
3	A Hard Day's Night
4	Beatles for Sale
5	Help!
6	Rubber Soul
7	Revolver
8	Sgt. Pepper's Lonely Hearts Club Band
9	Magical Mystery Tour
10CD1	The Beatles
10CD2	The Beatles
11	Abbey Road
12	Let It Be

Table A.2: Song list of Queen, Zweieck, and Carol King.

Artist	**Track**
Zweieck	Mr Morgan
Zweieck	Akne
Zweieck	Zuhause
Zweieck	She
Continued on next page	

Artist	Track
Zweieck	Duell
Zweieck	Erbauliche Gedanken Eines Tobackrauchers
Zweieck	Santa Donna Lucia Mobile
Zweieck	Tigerfest
Zweieck	Blass
Zweieck	Ich Kann Heute Nicht
Zweieck	Rawhide
Zweieck	Liebesleid
Zweieck	Jakob Und Marie
Zweieck	Zu Leise Für Mich
Zweieck	Es Wird Alles Wieder Gut, Herr Professor
Zweieck	Andersrum
Zweieck	Spiel Mir Eine Alte Melodie
Zweieck	Paparazzi
Queen	Somebody To Love
Queen	Another One Bites The Dust
Queen	Play The Game
Queen	I Want To Break Free
Queen	Hammer To Fall
Queen	Bicycle Race
Queen	Fat Bottomed Girls
Queen	Good Old
Queen	Friends Will Be Friends
Queen	You’re My Best Friend
Queen	A Kind Of Magic
Queen	Crazy Little Thing Called Love
Queen	We Are The Champions
Queen	Who Wants To Live Forever
Queen	Seven Seas Of Rhye
Queen	We Will Rock You
Queen	Bohemian Rhapsody
Queen	I Want It All
Queen	Don’t Stop Me Now
Queen	Save Me

Continued on next page

Artist	Track
Carole King	I Feel The Earth Move
Carole King	It's Too Late
Carole King	Beautiful
Carole King	You've Got A Friend
Carole King	Home Again
Carole King	Way Over Yonder
Carole King	So Far Away

Table A.3: Quaero Dataset.

Artist	Album	Track
A ha		Take On Me
Patrick Hernandez		Born to be alive
George Michael		Careless Whisper
Dolly Parton	Coat Of Many Colors	Travelling Man
Run DMC		It's like
Eminem	The Eminem Show	Cleanin Out my Closet
Enya	Shepherd Moons	Caribbean blue
Shack	HMS Fable	Natalies Party
CoCo Lee	Just No other Way	Do You Want My Love
Vangelis		Conquest of Paradise
Mariah Carey		Without you
The Beatles	Magical Mystery Tour	Baby Youre A Rich Man
Phil Collins		Another Day in Paradise
The Beatles	Abbey Road	Sun King
Bananarama		Venus
Offspring	Smash	Come out and play
FR David		Words
La Bouche		Be My Lover
Ace of Base		All that she wants
Queen	A Night at the Opera	Lazing on a sunday afternoon
Dusty Springfield	Dusty in Memphis	Son of a Preacher Man
Cher		Believe

Continued on next page

Artist	Album	Track
Santa Esmeralda		Don t Let Me Be Misunderstood
Aqua		Barbie Girl
Culture Beat		Mr Vain
Whitney Houston		I Will Always Love You
Kiss		I Was Made For Loving You
Enigma		Sadeness
Sean Kingston	Sean Kingston	Take You There
Jordin Sparks	Jordin Sparks	No Air
4 Non Blondes		Whats up
Crash Test Dummies		Mmm Mmm Mmm Mmm Mmm
Lil Mama	Voice of the Young People	Shawty Get Loose
Outkast	Aquemini	Chonkyfire
George Michael		Careless Whisper
Kylie Minogue		Cant Get You Out Of My Head
Leona Lewis	Spirit	Bleeding Love
Soul Asylum		Runaway train
DAngelo	Brown Sugar	Higher
East 17		Its Alright
Dillinger	Cocaine	I Thirst
Nickelback		How You Remind Me
Puff Daddy Feat Faith Evans		I'll Be Missing You
Pop Tops		Mamy Blue
The Beatles	A Hard Days Night	Tell Me Why
Bobby McFerrin		Dont worry Be happy
Chris Brown	Exclusive	Forever
U2	The Joshua Tree	With or without you
Womack and Womack		Teardrops
Twenty Fingers		Short Dick Man
Rolling Stones		Angie
Spice Girls		Wannabe
The Beatles	Beatles For Sale	I'm A Loser
Bon Jovi		Its My Life
Will Smith		Men In Black
Mariah Carey	E MC2	Touch my body

Continued on next page

Artist	**Album**	**Track**
Shakira		Whenever Wherever
Mamas and Papas		Dream a Little Dream of Me
Carl Douglas		Kung Fu fighting
Metallica		Nothing Else Matters
Eminem		Without Me (Radio Edit)
Nirvana	In Utero	Rape me
The Beatles	Abbey Road	Maxwell's Silver Hammer
Nena		99 Luftballons
Haddaway		What is love
Harold Faltermeyer		Axel F
Modern Talking		You re My Heart You re My Soul
Enya		Orinoco Flow
Fall out boy	Infinity on High	this aint a scene its an arms race
Xzibit		X
The Beatles	Magical Mystery Tour	Your Mother Should Know
Ricky Martin		The Cup Of Life
Fools Garden		Lemon Tree