

STRUCTURAL ANALYSIS AND SEGMENTATION OF MUSIC SIGNALS

A DISSERTATION SUBMITTED TO THE DEPARTMENT OF TECHNOLOGY OF THE
UNIVERSITAT POMPEU FABRA FOR THE PROGRAM IN COMPUTER SCIENCE AND
DIGITAL COMMUNICATION IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

DOCTOR PER LA UNIVERSITAT POMPEU FABRA

Bee Suan Ong
2006

© Copyright by Bee Suan Ong 2006
All Rights Reserved

DOCTORAL DISSERTATION DIRECTION

Dr. Xavier Serra
Department of Technology
Universitat Pompeu Fabra, Barcelona

This research was performed at theMusic Technology Group of the Universitat Pompeu Fabra in Barcelona, Spain. Primary support was provided by the EU projects FP6-507142 SIMAC <http://www.semanticaudio.org>.

Abstract

Automatic audio content analysis is a general research area in which algorithms are developed to allow computer systems to understand the content of digital audio signals for further exploitation. Automatic music structural analysis is a specific subset of audio content analysis with its main task to discover the structure of music by analyzing audio signals to facilitate better handling of the current explosively expanding amounts of audio data available in digital collections.

In this dissertation, we focus our investigation on four areas that are part of audio-based music structural analysis. First, we propose a unique framework and method for temporal audio segmentation at the semantic level. The system aims to detect structural changes in music to provide a way of separating the different “sections” of a piece according to their structural titles (i.e. intro, verse, chorus, bridge). We present a two-phase music segmentation system together with a combined set of low-level audio descriptors to be extracted from music audio signals. Two different databases are used for the evaluation of our approach on a mainstream popular music collection. The experiment results demonstrate that our algorithm achieves 72% accuracy and 79% reliability in a practical application for identifying structural boundaries in music audio signals.

Secondly, we present our framework and approach for music structural analysis. The system aims to discover and generate unified high-level structural descriptions directly from the music signals. We compare the applicability of tonal-related features generated using two different methods (the Discrete Fourier Transform and the Constant-Q Transform) to reveal repeated patterns in music for music structural analysis. Three different audio datasets, with more than 100 popular songs in various languages from different regions of the world, are used to evaluate and compare the performance of our framework with the existing system. Our approach achieves overall precision and recall rates of 79% and 85% respectively for correctly detecting significant structural boundaries in the music signals from three datasets.

Thirdly, we identify significant representative audio excerpts from music signals based on music structural descriptions. The system extracts a short abstract that serves as a thumbnail of the music and generates a retrieval cue from the original audio files. To obtain valid subjective evaluation based on human perception, we conducted an online listening test using a database of 18 music tracks comprising popular songs from various artists. The results indicate strong dependency between subjects' musical backgrounds and their preference for specific approaches in extracting good music summaries. For song title identification purposes, both evaluated objective and subjective results are consistent.

Fourthly, we investigate the applicability of structural descriptions in identifying song versions in music collections. We use tonal features of the short excerpts, extracted from the audio signals based on our prior knowledge of the music structural descriptions, to estimate the similarity between two pieces. Finally, we compare song version identification performance of our approach with an existing method [Gómez06b] on the same audio database. The quantitative evaluation results show that our approach achieves a modest improvement in both precision and recall scores compared to previous research work. To conclude, we discuss both the advantages and the disadvantages of our proposed approach in the song version identification task.

Acknowledgements

I would like to thank my supervisor, Dr. Xavier Serra, for giving me the opportunity and financial support to work in the Music Technology Group. He has introduced me to the field of music content processing. This work would not have been possible without his help.

I would also like to thank my research project manager, Perfecto Herrera, for his limitless patience in discussing this work with me regardless of his many other commitments. During my study, I have learned a lot from his helpful insights and valuable suggestions.

I want to thank all my colleagues from SIMAC research group, Emilia Gomez, Fabien Gouyon, Enric Guaus, Sebastian Streich, Pedro Cano and Markus Koppenberger for providing a very rare kind of stimulating and supportive environment. I am very grateful to their excellent technical advice and insight throughout this work. Without their presence, I would still be wandering. For assistances in manually annotating our used test data, I would also like to thank Edgar Barroso and Georgios Emmanouil. Without them, I will not have sufficient test data to conduct any kind of experiments.

For moral and emotional support, I would like to thank my dear big sister, Dr. Rosa Sala Rose, for her love, encouragement and for creating such a fun and enjoyable experience throughout my stay in Spain.

Finally, I wish to thank my family back in Malaysia for their unlimited support for my study trip to Spain. Particularly, I wish to express my profound gratitude to my parents, whose love, support and teachings made me that way I am.

Contents

Abstract.....	iv
Acknowledgement.....	vi
1 Introduction.....	1
1.1 Motivation and Goal.....	1
1.2 Multimedia Content Analysis	4
1.3 Music Audio Content Analysis.....	5
1.4 Music Structural Analysis.....	5
1.5 Applications.....	6
1.6 Scope.....	7
1.7 Summary of the PhD work.....	8
1.8 Thesis Outline.....	9
1.9 Description of Test Databases Used in Each Chapter.....	11
2 Literature Review.....	13
2.1 Introduction.....	13
2.2 Related Work in Automatic Music Structural Analysis.....	15
2.2.1 Audio Features.....	15
2.2.1.1 Timbre-related features.....	16
2.2.1.2 Harmonic and Melody-related features.....	18
2.2.1.3 Dynamics-related features.....	21
2.2.2 Feature Extraction Approach.....	22
2.2.3 Audio Segmentation.....	23
2.2.3.1 Model-free Segmentation.....	25
2.2.3.2 Model-based Segmentation.....	27
2.2.4 Music Structure Discovery.....	27

2.2.4.1 Self-Similarity Analysis.....	28
2.2.4.2 Dynamic Programming.....	32
2.2.4.3 Clustering.....	33
2.2.4.4 Hidden Markov Modeling.....	35
2.3 Discussion.....	37
2.4 Summary.....	39
 3 Semantic Audio Segmentation.....	41
3.1 Approach.....	42
3.1.1 Feature Extraction.....	43
3.1.2 Phase 1 – Rough Segmentation.....	45
3.1.3 Phase 2 – Segment Boundaries Refinement.....	55
3.2 Evaluation.....	57
3.2.1 Datasets.....	58
3.2.2 Procedure.....	58
3.2.3 Results and Discussion.....	59
3.3 Summary.....	65
 4 Music Structural Analysis Based on Tonal Features.....	67
4.1 Approach.....	68
4.1.1 Feature Extraction.....	70
4.1.2 Similarity Measurement.....	73
4.1.3 Pre-processing.....	76
4.1.4 Repetition Detection (Listing the repeated sections)	81
4.1.5 Integrating the Repeated Sections.....	83
4.1.6 Repetitive Segments Compilation.....	86
4.1.7 Boundaries Adjustment based on Semantic Audio Segmentation.....	87
4.1.8 Modulation Detection.....	90
4.1.9 Structural Description Inference.....	93
4.2 Evaluation.....	94
4.2.1 Data set.....	94
4.2.2 Quantitative Performance.....	95
4.2.3 Results and Discussion.....	95
4.3 Summary.....	104
 5 Identifying Representative Audio Excerpts from Music Audio.....	105

5.1	Audio Excerpt Identification and Extraction.....	106
5.1.1	First-30-seconds.....	106
5.1.2	Most-repetitive.....	107
5.1.3	Segment-to-Song.....	107
5.2	Evaluation.....	109
5.2.1	Subjects.....	109
5.2.2	Datasets.....	109
5.2.3	Web Interface and Listening Test Procedures.....	110
5.3	Observations and Results.....	116
5.4	Summary.....	121
6	Structural Descriptions for Song Version Identification in Music Collections.....	123
6.1	Short Summary Approach.....	124
6.1.1	Repetitiveness Emphasized.....	124
6.2	Evaluation.....	129
6.2.1	Dataset.....	129
6.2.2	Quantitative Measurements.....	129
6.2.3	Results.....	130
6.2.4	Discussion.....	134
6.3	Summary.....	135
7	Conclusions and Future Work.....	137
7.1	Summary of Contributions.....	137
7.2	Conclusion.....	140
7.3	Future Work.....	141
7.4	Final Thoughts.....	143
	Bibliography.....	145
	Appendix A Glossary.....	155
	Appendix B Details on Audio Database used in Chapter 3	157
	Appendix C Details on Audio Database used in Chapter 4.....	160

Appendix D Details on Audio Database used in Chapter 5.....	164
-------------------------------------------------------------	-----

List of Figures

<u>Figure</u>		
2.1	Illustration of categories of feature attributes.....	16
2.2	The pitch helix model.....	19
2.3	Illustration of long-term segmentation.....	24
2.4	Similarity matrix and novelty score computed from an audio excerpt from the soundtrack of <i>Beauty and the Beast</i> . The MFCC derivatives were used as low-level features.....	26
2.5	Pseudo code depicts the chorus detection procedure by Goto's RefraiD method.....	31
2.6	Dynamic Programming Scoring matrix, M_i	33
2.7	A 4-state ergodic hidden Markov model.....	35
3.1	Overview framework of our approach.....	43
3.2	Two-dimensional cosine similarity plot computed from the song entitled <i>When I Get Home</i> using MFCC features.....	46
3.3	Examples of how dilation and erosion work with the shaded structuring elements show the origin element.....	47
3.4	The properties of one-dimensional signal, A_x , with its structuring element, B_i , in defined in expressions 3.3 and 3.4.....	48
3.5	The properties of one-dimensional signal, A_x , with its structuring element, B_i , in defined in expressions 3.5 and 3.6.....	49
3.6	The opening operation of morphological filter on one-dimensional binary signal.....	50
3.7	The 'Open-Close' and 'Close-Open' operations of morphological filter on one-dimensional binary signal.....	51
3.8	Similarity representation before morphological operation (top) versus similarity representation after 'Close-Open' operation.....	52

3.9	Similarity representation after ‘Open-Close’ operation.....	53
3.10	Distance matrix representation obtained from the multiplication between ‘Open-Close’ and ‘Close-Open’ filter results.....	53
3.11	Detected boundaries candidates yielded by segment detection process in phase 1.....	54
3.12	The (dis)similarity representations between segments detected in phase 1.....	57
3.13	The novelty measures computed from the (dis)similarity representations between segments.....	57
3.14	An example of measuring segmentation performance with a tolerance deviation presented as shaded area (top transcription: ground truth segments; bottom transcription: detected segment). Circled segments mark the outliers of the correctly detected segments.....	59
3.15	The precision, recall and F-measure scores obtained for all songs in The Beatles’ test set with a tolerance deviation ± 3 seconds.....	60
3.16	The precision, recall and F-measure scores obtained for all songs in Magnatune’ test set with a tolerance deviation ± 3 seconds.....	61
3.17	Manually labelled segment boundaries (top) and segment boundaries detected by our proposed algorithm (middle) with time position information (below) for SongID-35 entitled <i>Words of Love</i> . The label VerseS means an instrumental solo playing the verse. Labels are not yet assigned by the algorithm. Circled segments mark the outliers of the correctly detected segments.....	61
3.18	The segmentation performance using various combinations of audio descriptors.....	63
3.19	Overview block diagram of our approach with the application of beat detection algorithm.....	64
3.20	The segmentation performance with and without the application of beat detection from The Beatles’ test set.....	64
3.21	The histogram of the average inter-beat interval detected in all songs in The Beatles’ database.....	65
4.1	Overview framework of our music structural description system	69
4.2	Self-similarity matrices of three notes, which include B4 played by the bassoon (<i>B_B4</i>), B4 by the clarinet (<i>Cl_B4</i>), and C5 by the bassoon (<i>B_C5</i>), using different Constant-Q feature vectors.....	72
4.3	General diagram for computing pitch class distribution features.....	72
4.4	Two-dimensional similarity plot of The Beatles’ song entitled <i>I’m a Loser</i>	75

4.5	Self-similarity matrices of three notes, which include B4 played by the bassoon (<i>B_B4</i>), B4 by the clarinet (<i>Cl_B4</i>), and C5 by the bassoon (<i>B_C5</i>), using difference distance measures.....	76
4.6	The time-lag matrix, L , for songs “I’m a loser” by The Beatles with its x-axis refers to the lag and y-axis refers to the time.....	77
4.7	Flow chart illustrates the iterative binarization process.....	77
4.8	An example of binarized time-lag matrix.....	79
4.9	Binarized time-lag matrix: before (upper) and after (below) applying morphological filtering operations.....	81
4.10	The possibility of containing repetition, $P_r(l,t)$, corresponds to each lag.....	82
4.11	Pseudo code outlines the line segments search algorithm.....	83
4.12	Detected repetitions correspond to the ground truth annotation of <i>A Hard Day’s Night</i>	84
4.13	The correlation between selected segment with pre-processed HPCP features, $v(n)$. Cross-circled marks the selected local minima based on the computed distances with a predefined threshold.....	85
4.14	Pseudo code outlines the line segment refinement algorithm.....	86
4.15	Repetitive segments compilation process with generated new labels.....	87
4.16	The output structural descriptions of the song entitled <i>All I’ve Got To Do</i>	88
4.17	The song example entitled <i>All I’ve Got To Do</i> with the alteration of line segments according to the information provided by semantic segmentation.....	90
4.18	The undetected modulated “refrain” segments within the song entitled <i>I Am Your Angel</i>	91
4.19	The correlation between the segment labeled A with transposed feature vectors, $V_{shift-semitone}(n)$. Circles mark the selected local minima as relevant modulated segments.....	92
4.20	The output structural descriptions of our proposed algorithm with the same song example given by Figure 4.15.....	93
4.21	Labeling integration procedure.....	93
4.22	Music structural descriptions from the song entitled <i>A Hard Day’s Night</i> , with various predefined d parameter settings.....	94
4.23	Precision measures of segmentation results (through structural analysis) with four different tonal-related descriptors using BeatlesMusic	96

4.24	Recall measures of segmentation results (through structural analysis) with four different tonal-related descriptors using BeatlesMusic	96
4.25	Evolution of recall and precision rates of HPCP with respect to the tolerance deviation (sec) for the different pitch class distribution features using ChaiMusic	97
4.26	Evolution of recall and precision rates of HPCP with respect to the tolerance deviation (sec) for the different pitch class distribution features using WorldPop	97
4.27	The segmentation performance (with a tolerance deviation of ± 3 seconds) on each song in WorldPop.....	98
4.28	The average of total F-measures obtained from each song in WorldPop along the twelve considered tolerance deviations.....	99
4.29	The SongID-12 entitled <i>I Say A Little Prayer</i> with two annotations.....	99
4.30	The song example entitled <i>Please Mister Postman</i> , where true negatives occur when different segments contains quite an identical temporal evolution of tonal descriptions (or chord progression in musical term).....	101
4.31	The segmentation evaluation results obtained using Euclidean distance (rightmost) versus Cosine distance (leftmost) using BeatlesMusic based on HPCP descriptors	102
4.32	The segmentation performance with and without the application of semantic audio segmentation on our proposed structural description algorithm using BeatlesMusic.....	102
4.33	Segmentation performance of various window lengths applying to morphological filter with a tolerance deviation of ± 3 seconds using BeatlesMusic.....	103
5.1	Segment-to-song distance computation.....	108
5.2	Introduction page of our online listening test.....	111
5.3	Subject registration page.....	112
5.4	The evaluation page.....	114
5.5	Help page for Question-2.....	115
5.6	Help page for Question-4.....	115
5.7	Feedback page.....	116
5.8	Subjects' age histogram.....	117
5.9	The evaluated song excerpts histogram according to subjects' musical background.....	117

5.10	The overall ratings of the subjective evaluation.....	118
5.11	The overall summary quality ratings for each approach used in identifying representative excerpts from music signals according to subjects' musical backgrounds.....	119
5.12	(Top) The song titles identification accuracy and (below) overall song summary quality obtained based on subjects' familiarity to the presented song excerpts	120
5.13	The recall effort required on each approach based on songs familiarity. (Description: x-axis represents various approaches and y-axis denotes recall effort.).....	121
6.1	The comparison of summaries between two songs.....	126
6.2	The circle of fifths geometry with major and minor modes. The major key for each key signature is shown as a capital letter on the outer circle whereas the minor key is shown as a small letter surrounded by the inner circle.....	126
6.3	The estimated minimum cost of the song summaries between a root query (song entitled <i>Imagine</i>) corresponding to 12 possible transpositions of its versions.....	128
6.4	The performances of version identification by using various numbers of short summaries of different lengths based on its average precision and recall measures	131
6.5	The performances of version identification: whole-song approach vs. short-summary approach based on its average precision and recall measures.....	131
6.6	Average F-measures of both approaches (short-summary and whole-song) in version identification according to the number of songs considered for a given user query.....	132
6.7	Average F-measures obtained for each retrieved song in the database with the considered first 10-retrieved songs for a given user query. Descriptions: Filled bars mark the cover songs of <i>Imagine</i> by different artists, whereas SongID-1 marked “*” denotes the root song, <i>Imagine</i> by The Beatles.....	133
6.8	Transitivity relationship between songs.....	134
7.1	An example of a sound visualization system coupled with music structure visualization and add-in segment playback functionalities.....	142
7.2	An example of finding song similarity system coupled with music structural visualization, add-in finding segment similarity and playback functionalities.....	144

List of Tables

Table

3.1	The list of audio descriptors for Phase 1 and Phase 2.....	45
3.2	Various combinations of the audio descriptors together with their labels appearing in Figure 3.17.....	62
4.1	The list of tonal-related descriptors generated using two different methods for structural description discovery	71
4.2	The different computation methods of the compared tonal-related features.....	74
5.1	Features grouping extracted from audio segments.....	108
5.2	Eighteen music pieces used in the online subjective evaluation.....	109
5.3	Objective evaluation results of song titles included in the excerpts generated using different approaches.....	118

Chapter 1

Introduction

This dissertation deals with structural description and segmentation of music audio signals. This work proposes systems intended to extract structural information from polyphonic audio recordings. We analyze different problems that arise when developing computational models that extract this musical information from audio, such as the extraction of significance features related to pitch-chroma for repetition identification and the selection of distance measure according to the used audio features. The goal of this chapter is to present the context in which this thesis has been developed, including the motivation for this work, the research context, some practical aspects related to automatic music structural and finally a summary of the work carried out and how it is organized along this document.

1.1. Motivation and Goal

As we enter a new advanced technology era, the explosion of multimedia content in databases, archives and digital libraries has caused some problems in efficient retrieval and management of this data content. Under these circumstances, automatic content analysis and processing of multimedia data becomes more and more important. In fact, content analysis, particularly content understanding and semantic information extraction, have been identified as important steps towards a more efficient manipulation and retrieval of multimedia content.

Grammar can be seen as the rules that are used to help a person to learn most written and spoken languages and to understand its meaning more quickly and efficiently. Like languages, music also uses grammatical rules in its various structural elements (i.e. harmony, rhythm, melody, etc.), even though the manner of constructing a music piece may vary widely from composer to composer and

from piece to piece. The importance of musical grammar used in constructing the underlying structure of a music piece can be seen through its applications in different domains. In the musicology domain, musicologists study musical grammar to analyze a piece of music. According to [Weyde03], music is highly structured and the perception and cognition of music rely on inferring structure to the sonic flow heard [Weyde03]. In the music perception and cognition domain, computational models [Lerdahl83] [Temperley01] have been developed by means of applying modern linguistic or universal grammar theories for music analysis to study the way humans perceive, process and mentally represent music. Seeing this, music structural analysis, which aims to compute a representation of the semantic content of a music signal through discovering the structure of music, is believed to be able to provide a powerful way of interacting with audio content (i.e. browsing, summarizing, retrieving and identifying) and facilitates better handling of music audio data.

In this research, we aim to provide an efficient methodology towards automatic audio-based music structure analysis. Here, we do not intend to infer the grammatical rules applied to the music composition. Instead, we aim to discover the structure of music through identifying the similarities or the differences of the overall structural elements, which musical grammars are applied to, within a composition. In addition, we attempt to identify “singular” within-song excerpts in popular music. We have focused our investigation in four areas that are closely related:

- (i) Semantic audio segmentation
- (ii) Music structure analysis and discovery
- (iii) Identification of representative excerpts of music audio
- (iv) Song version identification by means of music structural description

Semantic audio segmentation attempts to detect significant structural changes in music audio signals. It aims to provide direct access to different “sections” of a popular music track, such as the “intro”, “verse”, “chorus” and so on. Whereas music structure analysis and discovery, which is more of a pattern identification problem, aims to disclose and infer musical forms appearing in the acoustic signals. The ultimate goal of music structure discovery is the generation of complete and unified high-level descriptions of music. In contrast, the identification of representative excerpts of music audio aims to recognize the significant audio excerpts that represent a whole piece of music. The significant excerpts may consist of the most repetitive segments or even the most outstanding or “attention-grabbing” segments that are usually not repeated but are capable of leaving a strong impression on our minds. The goal behind representative excerpts identification is to generate a thumbnail or cue abstraction of the music that would give listeners an idea of a piece of music without having to listen to the whole piece. This would be very much useful in facilitating time-saving browsing and retrieval

of music, since it saves a considerable amount of time and thus speeds up the iterations. Burns [Burns87] provides a framework that categorizes the possible types of ‘hooks’ appearing in popular records into two main elements (i.e. textual and non-textual). Textual elements mainly consist of music structural elements (i.e. rhythmic, melodic, harmonic, etc.) whereas non-textual elements comprise of performance elements (i.e. tempo, dynamics, improvisation and accidentals) and production elements (i.e. sound effects, signal distortion, channel balance, etc.). Considering direct transcription of music structural elements, such as rhythmic and melodic elements, from polyphonic audio signals is infeasible with currently available technologies, so we simplify the representative excerpts identification task by taking into account the overall structural elements, without looking into each element in detail.

Finally, music structural description in song version identification endeavors to address the application issue of structural description in the music information retrieval context. The goal is to generate useful short excerpts from audio signals, which are based on the prior knowledge of the music structural information, to be used in the music retrieval system for finding different versions of the same songs. By using short audio excerpts instead of the whole song, it would allow the music retrieval system to search songs from a substantial amount of audio data within a tolerable time span and thus facilitate the retrieval task of the system.

It is important to note that in this study we do not deal with all kinds of music. Here, we are only interested in the structural analysis of pop music. Thus, other music genres, such as classical, jazz, ethnic and so forth, will not be included in our research. Music can be represented in two different manners: symbolic representation and acoustic representation. The symbolic representation is based on score-like notation of music. Thus, only a limited set of music material (e.g. pitches, note duration, note start time, note end time, loudness, etc.) is involved in music representation. Examples of such representation include MIDI and Humdrum [Selfridge97, Huron99]. The acoustic representation is based on the sound signal itself. Thus, acoustic representations can represent any sound in the natural world. Different storage formats and different lossy compression standards have led to different formats for acoustic representations of music. They include wav, au, or mp3-files to name a few. The task of automatic music structural analysis can be accomplished using either of these music representations. However, considering the prevalent usage of acoustic representation in representing popular music and that the task of converting acoustic representations to symbolic representations of music is still a currently open issue, we concentrate our study on dealing with acoustic representations.

1.2. Multimedia Content Analysis

With the rapid increase in electronic storage capacity and computing power, the generation and dissemination of digital multimedia content experiences a phenomenal growth. In fact, multimedia is pervasive in all aspects of communication and information exchange even through internet networking. Efficient management and retrieval of multimedia content have become the key issue especially for large distributed digital libraries, databases and archives. Traditional search tools are built upon the success of text search engines, operating on file names or metadata in text format. However these have become useless when meaningful text descriptions are not available [Cheng03]. Apparently, large indexing of multimedia content based on human efforts is very time consuming and may also lead to incoherent descriptions by different indexers and errors caused by carelessness. This causes problems when searching on improper indexed multimedia databases using text descriptions. Thus, a truly content-based retrieval system should have the ability to handle these flaws caused by text descriptions. So far, much research has been focusing on finding ways of analysis and processing to effectively handle these enormous amounts of multimedia content. In this context, multimedia content analysis, which aims to compute semantic descriptions of a multimedia document [Wang00], holds a tremendous potential.

The term “media” encompasses all modalities of digital content, such as audio, image, language. Video, which is used in entertainment, broadcasting, military intelligence, education, publishing and a host of other applications, represents a dynamic form of media. Digital video is a composite of image, audio, language and text modalities [Smith04]. So far, content-based analysis of video has been a fast emerging interdisciplinary research area. Prior video content-based analysis used physical features, such as colour, shape, texture and motion for frame characterization and later on scene recognition using similarity between frame attributes to study its content. Current video content-based analysis makes use of audio information included in video to facilitate better content descriptions [Zhu03] [Liu04]. The exploration of the significance of audio characteristics in semantic video content understanding has led audio content to be associated with video scene analysis, such as video segmentation, scene content classification and so forth, to facilitate easy browsing [Adam03]. In fact, audio content-based analyses are important processes in video characterization that aims to preserve and communicate the essential content of the video segments via some visual representation [Smith04]. Most characterization techniques use the visual stream for temporal segmentation and the audio stream is then analyzed for content classification [Nam97] [Wang00] [Pfeiffer01]. The development of MPEG-7 is an ongoing effort by the Moving Picture Experts Group to standardize such relevant features or metadata available for efficient characterization and descriptions of multimedia content. In this context, MPEG-7 holds a high potential in a variety of application

domains: large distributed digital libraries, digital and interactive video, multimedia directory services, broadcast media selection and multimedia authoring.

1.3. Music Audio Content Analysis

With the advance of compression technology and wide bandwidth of network connectivity, the existence of music downloading services on the internet blossoms. The availability of these services has made it possible for computer users to store many music files that he/she has only once or even never listened to. For instance, Napster, which offers over 700,000 songs, 70,000 albums and 50,000 artists to be downloaded for offline listening, is still adding new music to its database every day with new release from all of the four major music labels in the world, such as Sony/BMG, EMI, Warner Music Group and Universal Music Group. Apparently, the rapid increase of music collections has created difficulties for administrating these audio data. Retrieving a song without knowing its title from one of these huge databases would definitely be a difficult task. From this, we can see that the traditional way of music indexing and retrieval is no longer able to handle these huge databases. Thus, content-based analysis is believed to be suitable to facilitate efficient handling of these huge amounts of digital audio data. Similar to video content analysis, current music content analysis works focuses on generating semantic descriptions of the music that is contained in an audio file.

1.4. Music Structural Analysis

Music structure is a term that denotes the sounds organization of a composition by means of melody, harmony, rhythm and timbre. Repetitions, transformations and evolutions of music structure contribute to the specific identity of music itself. Therefore, laying out the structural plan (in mind or on a piece of paper) has been a prerequisite for most music composers before starting to compose their music. The uniqueness of music structure can be seen through the use of different musical forms in music compositions. For instance, western classical sonata music composers used the structural form known as sonata form, which normally consist of a two-part tonal structure, articulated in three main sections (i.e. exposition, development and recapitulation), to shape the music of a sonata. This is very different from the present popular music, which are much shorter in length and use much simpler structural forms. Thus, it is believe that the description of music structure, which subsume temporal, harmonic, rhythmic, melodic, polyphonic, motivic and textual information, is an important aspect in generating semantic descriptions from acoustic music signals. Comprehending such content descriptions may improve efficiency and effectiveness in handling huge music audio databases. Moreover, such structural description can also provide a better quality access and powerful ways of interacting with audio content, such as better quality audio browsing, audio summarizing, audio

retrieving, audio fingerprinting, etc., which would be very useful and applicable for music commercials and movie industries.

Recently music structural analysis has further extended its applications in the domain related to human cognition. Limitation of human memory makes us incapable to recall every single detail of all incidents that happen in our daily life. As human beings, we may only recall certain events, which have created a “strong” impression in our mind. The same happens with music, we do not recall the music that we hear in its entirety but through a small number of distinctive excerpts that have left an impression on our mind. It is usually the case that we only need to listen to one of those distinctive excerpts in order to recall the title for the musical piece or, at least, to tell if we have heard the song before. For instance, when a previously heard song is played halfway through on the radio, listeners are able to recognize the song without having to go through the whole song from the beginning until the end. According to psychology research, it is the retrieval cue in the music that stimulates us to recall and retrieve information in our memory [Schellenberg99]. Humans by nature own a remarkable object recognition capability. According to [Roediger05], people can often recognize items that they cannot recall. One example would be the experience of not being able to answer a question but then recognizing an answer as correct when someone else supplies it. In the music context, even musically untrained people are able to recognize or at least determine whether they have heard a given song before without much difficulty. Seeing the competency of structural analysis in distinguishing various structural elements of music directly from raw audio signals, much research related to music content description is currently focused on identifying representative musical excerpts of audio signals based on the derived structural descriptions.

In the following section, we review the potential of music structural analysis for a variety of application domains.

1.5. Applications

1. One of the primary applications for music structural analysis is the production of structural descriptors of music for music content exploitation. For instance, music can be classified according to the similarity or difference of its structural descriptions.
2. Music structural analysis has also some applications for facilitating time-saving browsing of audio files. Being able to provide higher semantic information from audio files would offer users some clues regarding where the structural changes in the audio occur (i.e. from “Intro” → “Verse” → “Chorus”, etc.). This would allow users to grasp the audio content through scanning the relevant segments. For example, an audio player with a functionality of allowing

users to skip from one section to another section would definitely reduce the browsing time of assessing large amounts of retrieved audio [Goto03b].

3. Repetition in music is one of the crucial elements in extracting a short abstract or generating a retrieval cue from the original audio files. Seeing that structural analysis holds a high potential in revealing the repeated patterns in music, it has also extended the applications for music summarization and thumbnailing.
4. Coupling music structural analysis functionality into music annotation tools would offer users with an initial audio annotation. In the case of manual annotation, annotators could profit from this initial annotation information from the system and make further adjustments or expansions of it. Without doubt, this would enhance the annotation processes.
5. The generation of higher-level semantic information of music audio may also provide an additional comparing dimension for music recommendation systems in finding music with similar characteristics. The systems can tailor users' preferences based on the simplicity or complexity of the music structure in the users' own collections.
6. Besides the usefulness in generating an abstract of the original audio files through music summarization, music structural analysis would also contribute in offering an interactive multimedia presentation that shows "key-frames" of important scenes in the music, allowing users to interactively modify the summary. For instance, users can create mega-tunes comprising a remix of all the choruses by their favourite artists.
7. Finally, automatic music structural analysis may serve as a valuable tool for computer support of most types of music, especially those not having scores at all or using non standard types of notation. Research work by Nucibella et al. [Nucibella05] shows an example of how computer based music analysis facilitates musicological research.

1.6. Scope

In this work, we examine four tasks of music structural analysis: (i) semantic audio segmentation; (ii) music structure analysis and discovery; (iii) identification of representative excerpts of music audio signals; and (iv) music structural description in song version identification. Four separate systems have been developed to automatically perform each task. All of them accept music audio as input signals. The segmentation system outputs a text file in ASCII format, which indicates the detected segment boundaries with a temporal resolution of 0.01 sec. The music structural analysis system outputs the transcription files in the lab-file format (used by WaveSurfer¹, an open-source tool for sound visualizing and manipulation, to transcribe sound files). It is noted that one audio input will only yield one unity transcriptions file marking the beginning and ending time of the repeated sections

¹ <http://www.speech.kth.se/wavesurfer/>

together with their given labels indicating the (dis) similar repeated sections (ex: A, B, C, etc.) appearing in the music signal. Whereas for the representative excerpt identification system, each input signal will cause the system to output three extracted 30-second excerpts of the input signal. These three excerpts are extracted based on each approach that we are interesting in studying. For the song version identification system, two short excerpts are extracted for each audio input. The system outputs a text-file, which contains a list of song-IDs from the dataset (excluding the root query song-ID) for each song query. Song-IDs in the lists are sorted according to the increasing order of the minimum costs computed between each song in the database with the root query song.

1.7. Summary of the PhD work

In this work, we undertake a study analyzing musical structure with the aim of discovering the musical forms that appear in the music signals and generating a high-level description from it. With the discovered music descriptions, we aim to identify characteristics within song excerpts from the perspective of content-based analysis. Repetitions and transformations of music structures contribute to the specific identity for each music piece. Thus, we hypothesize that identification of these transformations and the generation of a semantic level of music structural description will significantly contribute to better handling of audio files. In addition, we also intend to demonstrate the applicability potential of high-level structural descriptions in music information retrieval contexts. We do not attempt to investigate all kinds of music (i.e. classical, jazz, ethnic, to name a few) but only focus on “pop” music. Unlike much previous work in structural analysis [Lerdahl83], we make no attempt in tackling this matter based on symbolic notated music data (i.e. MIDI) but instead base our work on the actual raw audio. Hence, we rely on the particular characteristics of audio features in music content to perform structural analysis of music audio signals.

Our work contributes in a number of areas in music audio retrieval. In the audio segmentation task, we present our approach to detect the significant structural changes in audio content. In order to extract content descriptions that are significant in describing structural changes in music, we propose a combination set of low-level descriptors computed from audio signals. In addition, we also introduce the application of image processing filtering techniques for facilitating better segment boundaries detection. Finally, we use test database, which consists of popular music from different artists, to evaluate the efficiency of our proposal. The quantitative evaluation shows that our proposed approach achieves as high as 72% accuracy and 79% reliability in correctly identifying structural boundaries in music audio signals.

In music structural analysis and discovery tasks, we further improve previous research work in chorus identification [Goto03a] to produce a complete and unified high-level structural description

directly from music signals. We propose the use of timbre-based semantic audio segmentation to rectify the common boundaries inaccuracies, which appear in music structural descriptions caused by dependence on single tonal-related features to discover musical structure from acoustic signals. We also tackle the phenomenon of transposition within a piece of music by means of modifying the extracted tonal-related features. In addition, we propose the integration of timbre-based semantic audio segmentation into our system to rectify the boundary inaccuracies caused by the system's dependency on only tonal-related features for discovering structure in music. We then compare our segmentation performance with a previous method described in [Chai03c] to evaluate the efficiency of our proposal, and it shows improvement with respect to the overall performance.

In identifying representative audio excerpts of music, we take into consideration the potential of other possible approaches in capturing the specific features of the ‘gist’ in music instead of simply pursuing the present literature that mainly accentuates that repetitiveness of audio excerpts in the identification task. In addition, we conduct an online listening test to achieve some subjective evaluation regarding the quality of the extracted segments from various approaches, based on human perception. In our subjective evaluation based on human perception, our results show that our proposed approach is one of the most useful for song title identification compared to the rest of our studied methods for representative excerpts identification (i.e. first-30-segment approach and most-representative approach).

In song versions identification, we introduce a unique concept of using short representative excerpts from music to retrieve different song versions of the same songs. Here, we present our approach as to how to extract short excerpts from the audio signals based on structural descriptions of music for song versions identification. Finally, we use a song database, which consists of 90 versions of 30 different popular songs, to justify the feasibility of our proposed concept. Our quantitative results demonstrate an evident improvement in accuracy and time-saving factors for the song version identification task.

1.8. Thesis Outline

The remainder of this work is organized in the following manner.

Chapter 2 reviews related literature related to automatic music structural analysis. We include in this chapter a discussion regarding the pros and cons of each approach for discovering the structure of music as found in the literature.

Chapter 3 introduces our approach for semantic audio segmentation corresponding to the structural changes in music. It begins by giving an outline of our proposed method and this is followed by its full description. This chapter includes quantitative evaluation of the system's performance based on a test set. All experiments involve the use of polyphonic music from audio recordings of popular songs.

Chapter 4 presents our approach for music structural analysis and unified music description generation. It starts with giving a brief profile of our approach and is then followed by detailed descriptions of our approach. This chapter considers different test sets to assess the segmentation performance of our proposed system besides making comparisons with the existing system as well. The final section includes some discussion with regards to specific issues not solved by our system.

Chapter 5 attempts to identify representative excerpts in music audio signals. This chapter first lays down the framework of our method. It is then followed by a detailed description of our approach. We examine its performance based on different assumptions used in identifying representative audio excerpts through an online listening test. The test data includes popular songs from various artists. The final section of this chapter includes a discussion of the obtained subjective evaluation results based on human perception.

Chapter 6 investigates the applicability of structural descriptions for song version identification in music collections. This chapter begins with a brief introduction to different approaches. It is then followed by a full description of how the audio excerpts are extracted based on music structural descriptions for each approach. This chapter includes quantitative evaluations based on a test set consisting 90 versions from 30 different songs of popular music. The result observation section comprises quantitative comparisons among different approaches, including the one reported in the recent research work [Gómez06b]. The final section of this chapter discusses the shortcomings of our proposed approach in the song version task.

Finally, Chapter 7 draws conclusions and examines potential directions for future work in this area.

1.9. Description of Test Databases Used in Each Chapter

In this thesis, some chapters contain test databases that are used for evaluation purposes. Listed below are the used databases corresponding to their related chapters. Please refer to Appendix B for the full details of the test database.

Chapter 3

- 54 songs from The Beatles (1962 – 1965);
- 27 pop songs from the Magnatune² database;

Chapter 4

- 56 songs from The Beatles 70s' albums referred to as BeatlesMusic;
- 26 songs by The Beatles from the years 1962-1966 referred to as ChaiMusic;
- 23 popular songs in various languages referred to as WordPop;

Chapter 5

- 18 popular songs from The Beatles' and other artists or groups;

Chapter 6

- 90 versions from 30 different songs (root query) of popular music as described in [Gómez06b];

² Magnatune official web page: <http://magnatune.com/>

Chapter 2

Literature Review

In this chapter, we present a review of the literature related to the topic of this thesis. It starts with a general overview of music structural analysis. Following this, we review the research that is directly related to music structural analysis. Current research works in music structural analysis can be classified into two main approaches: the audio-signal approach versus the symbolic representation approach. The audio-signal approach deals with the actual raw audio file whereas the symbolic representation approach deals with music symbolic notation data (e.g. MIDI). Here, we focus our literature review on the audio-signal approach rather than on the symbolic representation. Feature extraction is an indispensable process in music content analysis. Thus, we devote some space to present the different extracted features considered in the literature. Audio segmentation facilitates division of audio signals for further analysis. In fact, it seems to be an indispensable procedure in certain content-based analysis. Here, we review work relevant to segmenting audio signals for further structural analysis. Music structural discovery aims to the identification of representative excerpts of music is a key issue in this thesis. Thus, in the last section of the literature review, we focus on relevant approaches for the identification task and the pros and cons of each proposed approach.

2.1. Introduction

A piece of music can be divided into sections and segments at a number of different levels. Lerdahl and Jackendoff [Lerdahl83] proposed the term *grouping* to describe the general process of segmentation at all levels (and the multi-leveled structure that results). Grouping of musical elements plays an important role in the recognition of repeated patterns or “motives” in music. According to psychological experiments [Dowling73] [Deutsch80] [Boltz86], if a sequence of notes is being

perceived as a group, it should be more easily identified and recognized than other sequences. Much works in musical grouping have adopted the Gestalt principles of perception organization. These adopted Gestalt principles are such as similarity, proximity and continuity. Since 1970s, much computational models have been proposed focusing on deriving various aspects of structure by means of music analysis. These various aspects of structure are such as metrical structure [Lerdahl83] [Povet85] [Allen90] [Lee91] [Rosenthal92] [Large94] [Temperley99], melodic phrase structure [Tenney80] [Lerdahl83] [Baker89a] [Baker89b], contrapuntal structure [Huron89] [Marsden92] [Gjerdingen94] [McCabe97], harmonic structure [Winograd68] [Bharucha87] [Bharucha91] [Mazwell92], key structure [Longuet-Higgins71] [Holtzmann77] [Leman95] [Krumhansl90] [Vos96]. Melisma Music Analyzer³ presented by Temperley is the latest preference-rules-based computation system for music analysis that covers several aspects of music structure (i.e. metrical structure, harmonic structure, phrase structure, contrapuntal structure—the grouping of notes into melodic line—and key structure).

In the domain of human cognitive capacity, Lerdahl and Jackendoff [Lerdahl83] evolved a theory called *A Generative Theory of Tonal Music*. Their central purpose was to elucidate the organization that the listener imposes mentally on the physical signals of tonal music. In Lerdahl and Jackendoff's work, they presented a framework comprised a set of grammar rules operating on four kinds of hierarchical structure that models the listener's connection between the presented music surface of a piece and the structure he attributes to that piece. The four components are grouping structure (related to segmentation into motives, phrases and sections), metrical structure (defining hierarchy between strong and weak beats), time-span reduction (establishing the relative importance of events in the rhythmic units of a piece) and prolongation reduction (hierarchy in terms of perceived patterns of tension and relaxation).

So far, the above mentioned computational models for music structural analysis are mostly derived from analyzing western classical compositions. In addition, the analyses were mostly based on the symbolic representation of the music (i.e. MIDI). This is understandable because notated music, such as western classical repertoire, is normally written by a composer by means of symbolic representation in the form of a score and is then performed by a performer. Thus, the score is generally taken to represent the piece. Besides, the symbolic representation of the music often provides explicit information about infrastructural representation (e.g. meter, contrapuntal structure, phrase structure, etc.), thus it has become the main object of attention and analysis. However for other type of music, such as rock and pop, there is generally no score available [Temperley01]. Thus the representation of the music itself is likely a particular performance or music recordings. For this

³ <http://www.link.cs.cmu.edu/music-analysis/>

reason, the analysis of music structure for these kinds of music will have to be derived directly from the audio signal. So far, most research work in audio-based structural analysis has mainly focused on popular music. This is because the song structure of popular music very frequently consists of sections labeled as *intro*, *verse*, *chorus* or *refrain*, *bridge* and *outro*, which can be identified if one comprehends the characteristics of these sections in popular music. In popular song writing: *intro* (*outro*), as suggested by its name, indicates the introduction (conclusion) to a song. Thus, *intro* (*outro*) typically appears at the beginning (ending) of a song; *verse* is a lyrical melodic phrase in which the premise of the story of the song is introduced and developed through its lyrics; *chorus* or *refrain* is normally a repeating phrase that occurs at the end of each verse of a song. Generally its repeated phrase delivers the gist of the song; *bridge* is the transitional section connecting a verse and a chorus. Sometimes an instrumental section is added to the song structure, sometimes the bridge takes the form of an instrumental section. The instrumental section can be an imitation of a chorus or a verse or a totally different tune from any of these sections. Thus, by applying segmentation and pattern recognition techniques (e.g. such as self-similarity) to the acoustic music signals, one should be able to relate the different content-based repetitions in the physical signals to the song structure.

2.2. Related Work in Automatic Music Structural Analysis

In the following sections, we explore several research directly related to automatic audio-based music structural analysis in detail, with a particular focus on discovering structure descriptions. These related automatic structural analysis research works either form the basis for other studies (i.e. music summarization) or as the subject of study in itself. We begin with a discussion of audio features that are commonly used in music structural analysis literature. It is then followed by the review of audio segmentation approaches aiming at a better division of the audio signal for further structural processing. Finally, we discuss a variety of identification techniques to discover the structure of music for further exploitations.

2.2.1. Audio Features

In music content analysis, proper selection of audio feature attributes is crucial to obtain an appropriate musical content description. For music structural analysis, it is important to extract a kind of music representation that is able to reveal the structural information from the audio signal. Extracting symbolic score-like representation from music could be a possible way to complete the task of music structural analysis [Raphael02] [Klapuri03]. However due to the demanding constraints in extracting symbolic score-like representation from polyphonic music, this approach is practically infeasible. Instead, extracting low-level representations from the audio signal for musical content description is found to be an alternative way for accomplishing this task. The term low-level is usually

employed to denote features that are closely related to the audio signal, which are computed in a direct or derived way. Lately, low-level audio feature attributes, which describe the musical content of a sound signal, have been widely used in research works closely related to music structural analysis, such as audio segmentation or boundary detection, audio thumbnailing, chorus identification, music summarization and pattern analysis of music. In automatic audio-based music structural analysis related works, feature attributes are often computed on a frame-by-frame basis in order to obtain the short-term descriptions of the sound signal. The music signal is cut into frames of a fixed time length. For each of these frames, a feature vector of low-level descriptors is computed in either the time domain or the frequency domain. In accordance with the similarities and differences of the generated content descriptions, these feature attributes can be roughly classified into three groups: timbre-related features, melody-related features, and dynamics-related features. Figure 2.1 illustrates the overall taxonomy of features.

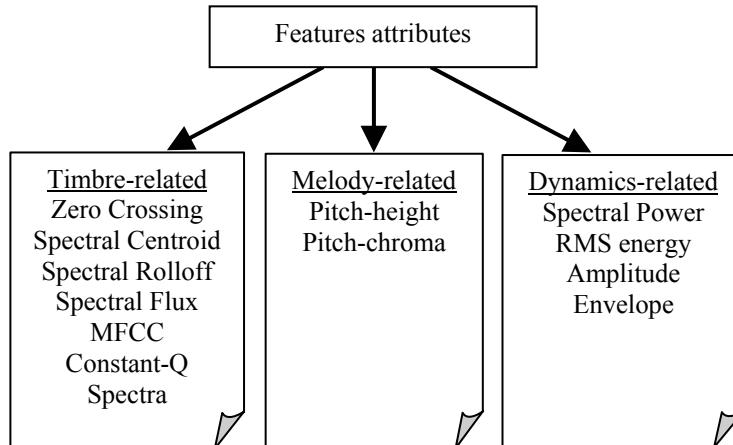


Figure 2.1. Illustration of categories of feature attributes

2.2.1.1 Timbre-related features

Timbre content descriptions are of general importance in describing audio. Most of the existing research work uses timbre content descriptions in order to differentiate music and speech besides music classification applications. Hence, many timbre-related features have been proposed in this research area [Tzanetakis99]. In fact, timbre-related features are the most widely used among the three groups mentioned above. So far, the most employed timbre-related features are:

Zero Crossings: A measure of the number of time-domain zero crossings within a signal. It gives an approximate measure of the signal's noisiness.

$$Z_t = \frac{1}{2} \sum_{n=1}^N |sign(x[n] - sign(x[n-1]))| \quad (2.1)$$

where *sign* function is 1 for positive $x[n]$ and -1 for negative $x[n]$ while t denotes the frame number.

Spectral Centroid: A representation of the balancing point of the spectral power distribution within a frame that is computed as follows:

$$SC = \frac{\sum_k kX[k]}{\sum_k X[k]} \quad (2.2)$$

where k is a correspond index to a frequency bin, within the overall estimated spectrum, and $X[k]$ is the amplitude of the corresponding frequency bin.

Spectral Rolloff: A measure of the frequency, below which 95 percentile of the spectral energy are accumulated. It is a measure of the “skewness” of the spectral shape – the value is higher for right-skewed distributions

$$SR = K, \text{ where} \quad (2.3)$$

$$\sum_{k < K} X[k] = 0.95 \sum_k X[k]$$

Spectral Flux (also known as Delta Spectrum Magnitude): A measure of spectral difference, thus it characterizes the shape changes of the spectrum. It is a 2-norm of the frame-to-frame spectral magnitude difference vector

$$SF = \|X[k] - X[k-1]\| \quad (2.4)$$

where $X[k]$ is the complete spectral magnitude of a frame.

MFCC, also called Mel-Frequency Cepstral Coefficients [Rabiner93]: A compact representation of an audio spectrum that takes into account the non-linear human perception of pitch, as described by the Mel scale. It is the most widely used feature in speech recognition. Currently, much research has focused in using MFCC to automatically discover the structure of music. [Aucouturier02] [Xu02] [Steelant02] [Logan00] [Peeters02] [Foote99] [Cooper02] [Kim06]. MFCC is particularly useful for analyzing complex music due to its low-dimensional, uncorrelated smooth version of the log spectrum, the ability to discriminate between different spectral contents [Steelant02] and to somehow discard

differences due to pitch evolution. MFCC calculation can be done through the following steps [Rabiner93]:

1. Convert signal into short frames
2. Compute discrete Fourier transform of each frame
3. Convert spectrum to the log scale
4. Mel scale and smooth the log scale spectrum
5. Calculate the discrete cosine transform (to reduce the spectrum to n^4 coefficients)

Constant-Q Spectra [Brown91]: A log frequency transformed of a fast Fourier transform. According to Brown, a constant Q transform can be calculated directed by evaluating:

$$X^{cq}[k_{cq}] = \sum_{n=0}^{N[k_{cq}]-1} w[n, k_{cq}] x[n] e^{-jw_{k_{cq}} n}, \quad (2.5)$$

where $X^{cq}[k_{cq}]$ is the k_{cq} component of the constant Q transform. Here $x[n]$ is a sampled function of time, and, for each value of k_{cq} , $w[n, k_{cq}]$ is a window function of length $N[k_{cq}]$. The exponential has the effect of a filter for center frequency $w_{k_{cq}}$. In practice, a constant Q transform can be implemented as a bank of Fourier filters of variable window width, where the centre frequencies of the constant Q filter banks are geometrically spaced. For musical applications, the calculation is often based on the frequencies of the equal tempered scaled with

$$w_{k_{cq}} = 2^{k_{cq}/12} w_{min} \quad (2.6)$$

for semitone spacing where w_{min} is the lowest center frequency of the used Fourier filters used.

2.2.1.2 Harmonic and Melody-related features

Melody, together with harmony, rhythm, timbre and spatial location makes up the main dimension for sound descriptions [Gómez03]. With the implicit information that it carries, melody plays an important role in music perception and music understanding. According to Selfridge-Field [Selfridge98], it is the melody that makes music memorable and enables us to distinguish one work from another. Current research in music content processing such as music transcription, melody similarity, melodic search, melodic classification and query-by-humming, works closely with melodic

⁴ The usual number of coefficients used for MFCC are less than 15.

information. So far, there are several ways of defining and describing a melody. Solomon [Solomon97] and Goto [Goto99, Goto00] define melody as a pitch sequence. While some others define music as a set of attributes that characterize the melodic properties of sound, a set of musical sounds in a pleasant order and arrangement etc. [Gómez03]. Among those viewpoints, melody as a pitch sequence would be the most appropriate representation for finding repetitions of music with the aim to discover music structure.

In pitch perception, humans recognize pitch as having two dimensions, which refer to pitch height and pitch chroma, respectively. Pitch chroma embodies the perceptual phenomenon of octave equivalence, by which two sounds separated by an octave (and thus relatively distant in term of pitch height) are nonetheless perceived as being somehow equivalent. Therefore, pitch chroma provides a basis for presenting acoustic patterns (melodies) that do not depend on the particular sound source. In contrast, pitch height varies directly with frequency over the range of audible frequencies. Hence, it provides a basis for segregation of notes into streams from separated sound sources. Within the music context, music psychologists represent pitch using a bi-dimensional model called the pitch helix model (as shown in Figure 2.2). In the helix model, the musical scale is wrapped around so that each circuit (marked red) is an octave [Warren03]. The pitch height representation moves vertically in octaves, and the pitch chroma representation determines the rotation position within the helix. The function of these two pitch dimensions is illustrated when the same melody is sung by a male or a female voice [Warren03]. In music notation, “A4” is used to give information regarding the pitch of a musical note in both dimensions (i.e. pitch height and pitch chroma). Alphabet, “A”, refers to pitch chroma while the number, “4”, denotes the pitch height.

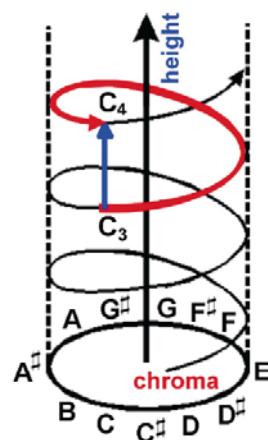


Figure 2.2. The pitch helix model.

In the music structural analysis and processing domain, melody-related features have been widely used in identifying repetitive patterns or representative excerpts of music. According to the dimension they focus on, we can consider two approaches in extracting melody-related features. The first one focuses on the pitch-height dimension. This approach uses features that carry pitch-height information to find repetitive patterns of music. Dannenberg and Hu, [Dannenberg02b] use this approach to estimate pitch and identify the note boundaries of monophonic music. The authors compute the correlation between the signal and a time-shifted version of it. Finally, the fundamental pitch is selected based on several heuristics rules. This approach is only applicable for single pitch monophonic music. However, for real-world polyphonic music with a complex mixture of pitches, extracting the predominant one is highly complicated and practically infeasible with current methods. Sound source separation, which aims to separate a sound mixture, could be a possible way to facilitate in extracting predominant pitch of music. However due to the present limitations of sound source separation technologies in performing precision separation of signals, extracted pitch height information from polyphonic music is still very unreliable.

The second approach focuses on the pitch-chroma dimension and thus uses features that carry pitch-chroma information. Pitch-chroma holds the information related to the harmony or the melodic content of music and it captures the overall pitch class distribution of music [Goto03a], the description it yields can be similar even if accompaniment or melody lines are changed to some degree. With this unique characteristic of pitch-chroma, there is no constraint of using this approach to analyze polyphonic music. In fact, the application of harmonic or melodic content-related information in music content processing is not a novel strategy. The pitch histogram proposed by Tzanetakis [Tzanetakis02] for measuring similarity between songs would be an example. Tzanetakis's pitch histogram is composed of a set of global statistical features related to the harmonic content. This set presents the most common pitch class used in the piece, the occurrence frequency of the main pitch class, and the octave range of the pitches of a song. In their research on the identification of representative musical excerpts research, several authors [Goto03a] [Dannenberg02a] [Birmingham01] [Bartsch01] [Bartsch05] have employed chroma-based vectors to find the repetitive patterns of music. A chroma-based vector is basically an abstraction of the time varying spectrum of audio. It is computed mainly through restructuring a sound frequency spectrum into a chroma spectrum. Octave information is discarded through folding frequency components in order to fall into twelve distinct chroma bins which correspond to the twelve pitch classes [Dannenberg02a]. Bartsch and Wakefield [Bartsch01, Bartsch05] perform autocorrelation to the chroma-based vector in order to identify the song extract, which holds the most repeated "harmonic structure". With a different formulation, Goto's [Goto03a] RefraiD method employs a 12-element chroma-based vector similar to the one that

is used in [Bartsch01], in order to analyze relationships between various repeated sections, and finally detecting all the chorus parts in a song and estimating their boundaries.

2.2.1.3 Dynamics-related features

In human auditory perception, loudness contrast captures listeners' ears. The musical term "dynamics", which refers to relative loudness or quietness measurement of the sound, holds a significant role in expressive musical structure formation. In music composition and music performance, artists use dynamics to emphasize and shape the structure of music. Current research studies in music expressive performance analyze dynamics behaviour to evaluate the expressiveness of the performance [Friberg04]. A real-time expressive music performance visualizing system, based on tempo and loudness spaces, has been built to help studying performance expressiveness. It depicts the dynamics and tempo behaviour of each performance done by different interpreters on the same piece of music [Widmer03]. Considering the significance of music dynamics in marking the occurrence of new music events, dynamics-related features have become unique and useful in music segmentation. When finding repetitions in music, proper identification of dynamics-based repetition boundaries is highly significant. So far, three dynamics-related features frequently appear in the existing work: Spectral Power, RMS and amplitude envelope.

Spectral power: For a music signal $s(n)$, each frame is weighted with a window. [Xu02] weights each frame signal with a Hanning window that is defined as $h(n)$:

$$h(n) = \frac{\sqrt{8/3}}{2} [1 - \cos(2\pi \frac{n}{N})] \quad (2.7)$$

where N is the number of the samples of each frame.

$$SP = 10 \log_{10} \left[\frac{1}{N} \left\| \sum_{n=0}^{N-1} s(n)h(n) \exp(-j2\pi \frac{n}{N}) \right\|^2 \right] \quad (2.8)$$

RMS energy [Tzanetakis99, Steelant02]: A measure of physical loudness of the sound frame

$$RMS = \sqrt{\frac{1}{N} \sum_{k=0}^{N-1} x[k]^2} \quad (2.9)$$

where N is the number of samples in each frame.

Amplitude Envelope: A description of the signal's energy change in the time domain. [Xu02] computes the signal envelope with a frame-by-frame root mean square (RMS) and a 3rd order Butterworth lowpass filter [Ellis94] with empirically determined cutoff frequencies.

2.2.2. Feature Extraction Approach

So far, there exist two approaches in using the above mentioned low-level feature attributes to obtain useful descriptions for music structure discovery: the static one and the dynamic one. The static approach computes low-level descriptions directly from the sound signal to represent the signal around a given time. Hence, in order to detect repetitive patterns in music, it is essential to find feature sequences with identical evolution. The dynamic approach, proposed by Peeters et al. [Peeters02], uses features that model directly the temporal evolution of the spectral shape over a fixed time duration. The difference between the two approaches is that the earlier one uses features that do not model any temporal evolution and only provide instantaneous representations around a given time window (i.e. only the successive sequence of the features models the temporal evolution of the descriptions).

Following the static approach, Steelant et al. [Steelant02] propose the use of statistical information of low-level features, instead of the features themselves, to find the repetitive patterns of music. These statistics are mainly the average and the variance of the instantaneous features over the whole signal. According to Steelant et al., global representations of the low-level features, which consist of their statistical information, can overcome the problem of very similar passages having different extracted coefficients, due to the large frame-step during feature extraction process. In their research to find the repetitive patterns of music, they use feature sets, which contain mean and standard deviation of MFCCs. Their algorithm, tested on a database of only 10 songs, showed a slight improvement when using the statistical information of the low-level features instead of using the frame by frame features.

On the dynamic approach side, Peeters et al. [Peeters02] compute dynamic-features by passing the audio signal, $x(t)$ through a bank of N Mel filters. Short-Time Fourier Transform (STFT) with window size L is then used to analyze the temporal evolution of each output signal $x_n(t)$ of the $n \in N$ filters. The transformed output, $X_{n,t}(w)$, models directly the temporal evolution of the spectral shape over a fixed time duration. According to Peeters et al., the window size that is used for STFT analysis determines the kind of music structure (i.e. *short-term* or *long-term*) that can be derived from the signal analysis. Even though this approach may greatly reduce the amount of used data, the advantage is only noticeable when one deals with a high dimensionality of feature attributes.

2.2.3. Audio Segmentation

Music structural discovery from audio signals was first inspired by the works on signal segmentation first developed in speech applications, such as “SpeechSkimmer” [Arons93], and were later adapted for musical applications. Thus, signal segmentation is closely associated with music structural discovery. In fact, signal segmentation, which facilitates partitioning audio streams into short regions for further analysis, is an indispensable process in music structure discovery. Finding appropriate boundary truncations is crucial for certain content-based applications, such as audio summarization and audio annotation. In this section, we will discuss different methods implemented for segmenting audio signals for later structural identification. In addition, we have grouped the methods according to their similarities and differences regarding implementation (i.e. model-free segmentation versus model-based segmentation).

In discovering structure of music, we can distinguish between two segmentation processes: short-term segmentation and long-term segmentation. Short-term segmentation (sometimes also called frame segmentation) is in fact a crucial primary step in content analysis description. This segmentation process normally partitions audio streams into fixed-length short regions for further analysis. These short regions may sometimes partially overlap. However due to arbitrary fixed resolution segmentation of audio streams may cause unnatural partitions, current development in this area has been the exploitation of high-level rhythmic descriptions, such as tempo tracking, beat or onset detection, to find natural segmentation points to improve the overall short-term segmentation performance [Maddage04] [Shao05] [Levy06a][Levy06b].

Maddage et al. [Maddage06] present an inter-beat segmentation known as beat space segmentation (BSS) to segment music signal into smallest note length with the use of onset detection. The authors first decompose music signal into 8 sub-bands corresponding to octaves of music scale. Using the similar method in [Duxburg02], the authors analyze both the frequency and energy transients of the sub-bands signals. An energy-based detector and frequency based distance measure are used on the upper (within the frequency range of 1025 to 22050 Hz) and lower (within the frequency range of 0 to 1024 Hz) sub-bands respectively. To detect both hard and soft onsets, the authors take the weighted summation of the detected onsets in each sub-band. By taking the autocorrelation over the detected onsets, the initial inter-beat length is estimated. Following this, dynamic programming approach [Navarro01] is applied to check for equally spaced beats patterns among the detected onsets and compute both the smallest note length and inter-beat length. Maddage et al. then segment music signal into smallest note length frames for later music structural analysis processes. It is important to note that beat space segmentation (BSS) is based on the assumptions that the time signature of an input song is 4/4 and the tempo of the song is constrained to between 30–240

quarter notes per minute. In addition, the tempo of the music is bounded to be roughly constant throughout the songs. Thus, for music signals which fail to fulfill these assumptions, the above mentioned segmentation approach is practically infeasible.

On the other hand, long-term segmentation aims to identify appropriate boundaries for partitioning the audio streams into sections. These sections comprise a non-fixed number of successive short regions being the output from earlier short-term segmentation processes (as shown in Figure 2.3), based on their feature changes. Hence, the partitions we obtain using long-term segmentation have a longer duration than those from short-term segmentation. Long-term segmentation assumes that the boundaries between two consecutive partitions should consist of abrupt changes in their features' contents. Meanwhile, the feature values of the signal inside each partition are supposed to vary little or slowly (i.e. are homogenous). Since appropriate boundary divisions are rather significant for music structure, this segmentation process holds an important role in automatic music structural analysis.

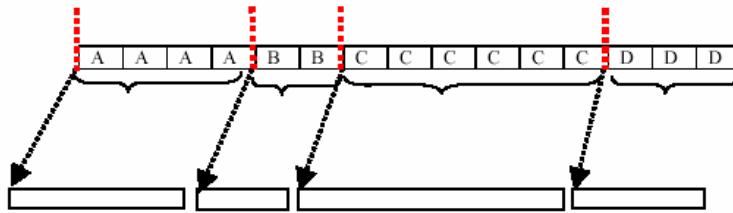


Figure 2.3. Illustration of long-term segmentation

Long-term segmentation strategies can be categorized into two groups, according to the similarities and differences in their implementations. Hence, we speak of model-free segmentation and of model-based segmentation. Model-free segmentation algorithms partition signals without requiring any training phase. An example of model-free long-term segmentation method used in automatic music structure analysis is similarity measures [Bartsch01] [Steelant02] [Cooper02] [Cooper03] [Goto03a] [Lu04] [Bartsch05]. In the case of model-based segmentation, a training phase is necessary in order to learn the models for segmenting. The model is built, by using a collection of examples, which correspond to the desired output from the segmentation algorithm, as training samples. Hidden Markov Models (HMM) [Logan00] [Aucouturier02] are an example of the model-based long-term segmentation method used in music structure analysis.

2.2.3.1 Model-free Segmentation

A widely used model-free segmentation technique takes advantage of (dis)similarity measures [Foote00] [Bartsch01] [Steelant02] [Cooper02] [Peeters02] [Cooper03] [Goto03a] [Lu04] [Bartsch05]. Foote [Foote99] first proposed the use of local self-similarity in spotting musically significant changes in music. It is done by measuring the distance between feature vectors using Euclidean distance or the cosine angle distance between the parameter vectors. The similarity matrix is a two-dimensional representation that contains all the distance measures for all the possibilities of frame combinations (as shown in Figure 2.4 - top illustration). As every frame will be maximally similar to itself, the similarity matrix will have a maximum value along its diagonal. In addition, if the distance measure is symmetric, the similarity matrix will be symmetric as well. With the use of a cosine angle distance, similar regions will be close to 1 while dissimilar regions will be closer to -1. According to Foote, by correlating a similarity matrix, S , with a checkerboard kernel, which is composed of self-similar values on either side of the centre points and of cross-similarity values between the two regions, along the diagonal of the similarity matrix, it yields the time instant of audio novelty $N(i)$, which is useful for identifying the immediate changes of audio structure. A simple 2x2 unit kernel, C , that can be decomposed into “coherence” and “anticoherence” kernels is shown in equation 2.10 below.

$$C = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (2.10)$$

Audio novelty can be represented by

$$N(i) = \sum_{m=-L/2}^{L/2} \sum_{n=-L/2}^{L/2} C(m, n) S(i + m, i + n) \quad (2.11)$$

where S denotes the similarity matrix, i denotes the frame number, and L represents the width of the kernel that is centered on 0,0. A visual rendering of a similarity matrix (top) (with a given grey scale value proportional to the distance measure) together with its corresponding novelty score (bottom) give a clear image display of the occurrences of different sections in audio, as shown in Figure 2.4.

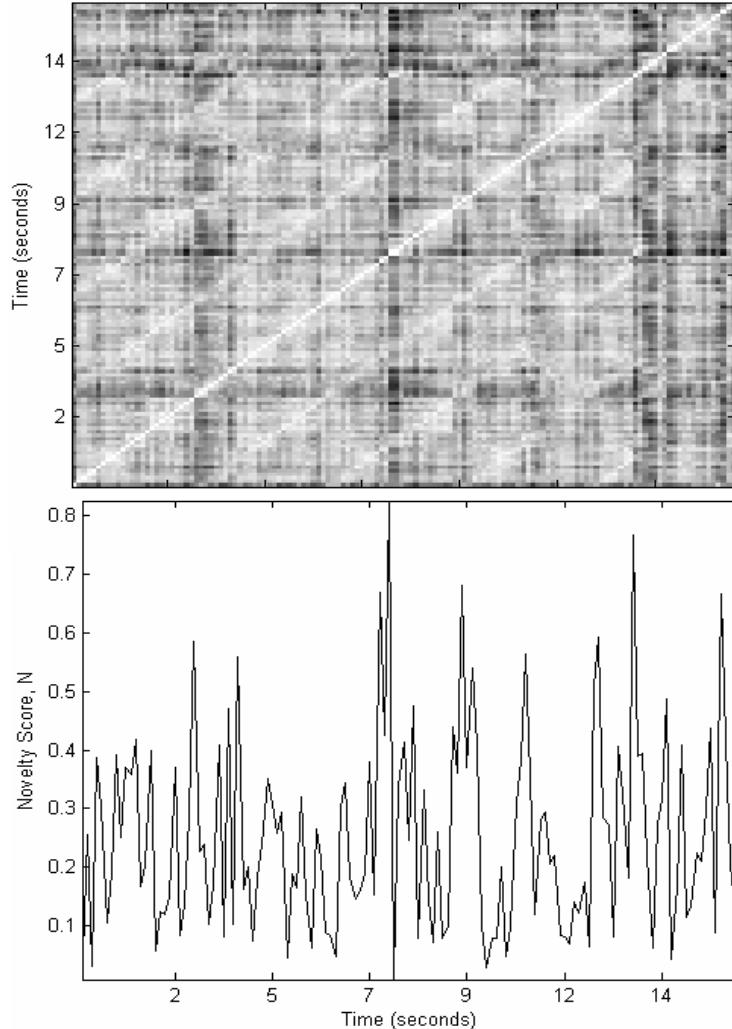


Figure 2.4. (Top) Similarity matrix and (bottom) novelty score computed from an audio excerpt from the soundtrack of *Beauty and the Beast*. The MFCC derivatives were used as low-level features

Given that novelty detection is based on the correlation process, the width of the kernel affects the resolution of the detection outcome. A small kernel, which detects novelty on a short time scale, is capable of identifying detailed changes in the audio structure such as the individual note events. On the other hand, a large kernel, which takes a broader view of the audio structure, compensates its coarse detection with a better identification for longer structural changes, such as music transitions and key modulations. According to Foote, A large kernel can be constructed by forming the Kronecker product of \mathbf{C} with a matrix of one and applying a window to smoothen the edge effects, for example,

$$\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & 1 \end{bmatrix} \quad (2.12)$$

Finally, segment boundaries are extracted by detecting peaks where the novelty score exceeds a local or global threshold. A binary tree structure is then constructed to organize the index points of the segment boundaries by the novelty score. Our semantic audio segmentation approach is mainly based on (dis)similarity measurement approach. Further details regarding our proposed semantic audio segmentation method will be presented in the next chapter.

2.2.3.2 Model-based Segmentation

Hidden Markov Models (HMM) [Rabiner86], a well-known technique in pattern discovery and speech processing, is an example of model-based segmentation used in the research aiming to identify representative musical excerpts. Aucouturier and Sandler [Aucouturier01] train a 4-state ergodic HMM with all possible transitions to discover different regions in music based on the presence of steady statistical texture features. In their experiments, they use the classic Baum-Welch algorithm to train the HMM. The algorithm optimizes the Gaussian mixture distribution parameters and the transition probabilities for each state of the model for the given training data. Finally, segmentation is deduced by interpreting the results from the Viterbi decoding algorithm for the sequence of feature vectors for the song. One of the two approaches used in Logan and Chu [Logan00] is another example of applying Hidden Markov Models in a long-term segmentation task. Since the segmentation and the identification processes are closely related, HMM is capable of integrating the segmentation and identification process into a unified process. In other words, it completes both tasks by using a single algorithm. The application of HMMs for solving identification tasks will be discussed in the following section.

2.2.4. Music Structure Discovery

Structural analysis seeks to derive or discover structural descriptions of music and provide a higher-level interactive way of dealing with audio files. Structural analysis research work such as, semantic audio segmentation [Chai03c], music thumbnailing [Bartsch05] [Chai03a] [Chai03b] [Aucouturier02], music summarization [Cooper03] [Xu02], chorus detection [Goto03a] [Bartsch01] and repeating patterns identification [Lu04], although carrying different titles, all shares the same goal of facilitating an efficient browsing and searching of music audio files. In fact, they are all built upon the identification of significant audio excerpts that are sufficient to represent a whole piece of music.

Hence, identifying the representative musical excerpts from music structure is the key issue here. There are different approaches, including those which are commonly used in pattern recognition and image processing. Here, we organize these approaches into four main groups: Self-similarity Analysis, Dynamic Programming, Clustering, and Hidden Markov Modeling, based on its differences and similarities. In the forthcoming subsections we discuss these approaches, including pros and cons of their specific algorithms.

2.2.4.1 Self-Similarity Analysis

In Section 2.2.3.1, we have seen how self-similarity facilitates in spotting musically significant changes in music for the audio segmentation task. Here, we are going to observe how self-similarity is exploited in discovering the structure of music. The occurrence of repetitive sections in the structure of music audio has led researchers to relate music audio structure with fractal geometry phenomena in mathematics. A few methods based on self-similarity have been employed for identifying representative musical excerpts. One of them is the two-dimensional self-similarity matrix [Foote00]. Seeing that self-similarity measurement is capable of expressing local similarity in audio structure, Bartsch and Wakefield [Bartsch01] use a restructured time-lag matrix to store the filtering results that are obtained through applying a uniform moving average filter along the diagonals of the similarity matrix, for the aim of computing similarity between extended regions of the song. Finally, they select the chorus section of music by locating the maximum element of a time-lag matrix based on two defined restrictions: (1) the time position index of the selection must have a lag greater than one-tenth of the song; (2) it appears before three-fourths of the way through the song.

Goto's [Goto03a] RefraiD method is another example of using time-lag similarity analysis in identifying representative musical excerpts from audio music. Goto also uses 2-dimensional plot representations having time-lag as their ordinate, in order to represent the similarity and the possibility of containing line segments at the time lag. With an automatic threshold selection method, which is based on a discriminant criterion measure [Otsu79], time-lags with high possibility of containing line-segments are selected. These selected time lags are then used to search on the horizontal time axis on the one-dimensional function for line segments using the same concept of the previous threshold selection method. After that, groups are used to organize those line segments, with each group consisting of the integration of the line segments having common repeated sections. The author then recovers the omitted line segments from previous line segment detection process through searching again the time-lag matrix using the line segment information of each group. Finally groups, which share a same section, are integrated into a singular group. The group integration process works by adding all the line segments belonged to the groups and adjusting the lag values. With the use of the corresponding relation between circular-shifts of the chroma vector and performance modulation,

Goto further improves the overall detection performance by tackling the problem in identifying modulated repetition. According to Goto, when an original performance is modulated by tr semitones upwards, its modulated chroma vectors satisfy,

$$\vec{v}(t) \doteq S^{tr} \vec{v}(t)' \quad (2.13)$$

where

$\vec{v}(t)$ = chroma vectors of original performance,

$\vec{v}(t)'$ = chroma vectors of modulated performance that is modulated by tr semitones upwards from the original performance,

S^{tr} = shift matrix defined by

$$\begin{pmatrix} 0 & 1 & 0 & \dots & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & & 0 & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & 1 & 0 \\ 0 & \dots & \dots & \dots & 0 & 1 \\ 1 & 0 & \dots & \dots & \dots & 0 \end{pmatrix} \quad (2.14)$$

By using this strategy, Goto computes twelve kinds of extended similarities using the shift matrix and chroma vectors of original performances in order to represent the modulation of twelves semitones upwards. Twelve kinds of extended similarity of each tr are defined as:

$$r_{tr}(t, l) = 1 - \frac{\left| \frac{S^{tr} \vec{v}(t)}{\max_c v_c(t)} - \frac{\vec{v}(t-l)}{\max_c v_c(t-l)} \right|}{\sqrt{12}} \quad (2.15)$$

where the denominator $\sqrt{12}$ is the length of the diagonal line of the 12-dimensional hypercube with edge length 1, $r_{tr}(t, l)$ satisfied $0 \leq r_{tr}(t, l) \leq 1$. For generating structural description of music (in Chapter 4), we will also tackle the issue of detecting modulations within a song.

For each kind of extended similarity, the above-mentioned process of listing and integrating the repeated sections is performed, with the exception that the threshold adjusted for the original performance vectors is used for modulation vectors as well. Goto later unfolds each line segment in each group to obtain unfolded repeated sections and λ_{ij} (its possibility of being chorus sections). Before the possibility λ_{ij} of each repeated section is used for later calculation for V_i (a total possibility of each group for being a chorus section), it is adjusted based on three heuristic assumptions,

- i. The length of the chorus section has an approximate range. If the length is out of range, λ_{ij} is set to 0.
- ii. Long repeated sections may correspond to a long-term repetition (e.g. the verseA, verseB and chorus) and it is likely that a chorus section is located near its end. Hence, if there exists a repeated section whose end is close to the end of another long repeated section (longer than 50 sec), its λ_{ij} is doubled.
- iii. Because a chorus section tends to have two half-length repeated sub-sections within its section, a section that has those sub-sections is likely to be the chorus section. If there is a repeated section that has those sub-sections in another group, half of the mean of the probability of those two sub-sections is added to its λ_{ij} .

Finally, the group with the highest possibility value of V_i is selected as the chorus section. Pseudo code in Figure 2.5 depicts the chorus detection procedure by Goto's RefraiD method. The experimental result reported in [Goto03a] shows quite a satisfactory result with 80 out of 100 songs have its chorus sections successfully detected. For the music structural discovery task, which aims to reveal structure descriptions from the music signal, the research work has only achieved partial success. This is because only specific music sections, such as chorus section in this case, are identified and labeled. Whereas there are still remaining sections left without being identified and labeled. Our approach in discovering and extracting music structural descriptions from music signal is highly inspired by [Goto03a]. We introduce further improvements on this method to offer a unity and high level of music structural description such that different sections that appear in the music signal are identified and labeled. Full detail regarding our approach will be presented in later Chapter 4.

Another research work by Lu et al. [Lu04] also uses self-similarity analysis approach to perform repeated pattern discovery and structural analysis from acoustic music data. Different from Goto's approach, Lu et al. use estimated rhythmic information such as, tempo period and length of musical phrase, to define the minimum length of a significant repetition in repeating pattern discovery and boundary determination. In addition, the authors utilize Constant-Q transform (CQT) (as explained in Section 2.2.1.1) in extracting features from music signals. With the extracted CQT features vectors, Lu et al. exploit structure-based distance to compute similarity measures between each pair of the musical frames. Structure-based distance is computed based on autocorrelation of the difference compared vector, with its number of lags corresponds to the features index number. The idea of using weighted autocorrelation coefficients is to reduce the distance measures' sensitivity for timbre difference. To facilitate the repetitions detection process, Lu et al. introduce morphological filtering technique, which will be explained in Chapter 3, to enhance significant repetitions lines and remove the short lines appear in the time-lag matrix. The authors employ the estimated length of a musical

phrase as presented in [Lu03] to define the length of the structuring element for morphological filtering process. Finally, with the detected repeated patterns, the author uses heuristic rules to infer music structure.

```
% for detecting non-modulated repetition
Compute time-lag matrix,  $r(t,l)$ 
Normalize  $r(t,l)$ 
Compute  $R_{all}(t,l)$  with normalized  $r(t,l)$ 
Set threshold,  $Th_R$  based on discriminant criterion measure
Execute function(detect_repetition)
Recover omitted line segments based on line segment information of each group
Integrate groups that share a same segment into a singular group

% for detecting the modulated repetition
For each semitone
    Compute modulated chroma vector,  $\vec{v}(t)$ , based on Equation (2.12)
    Compute  $r_{tr}(t,l)$  based on Equation (2.12)
    Normalize  $r_{tr}(t,l)$ 
    Compute  $R_{all}(t,l)$  with normalized  $r(t,l)$ 
    Execute function(detect_repetition)
    Recover omitted line segments based on line segment information of each
    group
    Integrate groups that share a same segment into a singular group
End

For each group
    Define  $\lambda_{ij}$  based on three heuristics rules
    Compute total possibility  $V_i$ 
End

If  $m = \underset{i \rightarrow group}{\operatorname{argmax}} V_i$ 
    Select  $m$  as chorus sections
End

function (detect_repetition)
{
    Let  $high\_peak = R_{all}(t,l)$  that is above  $Th_R$ 
    Let  $L_{high\_peak} =$  lag information of each  $high\_peak$ ,
    For each  $high\_peak$ 
        Search on the horizontal time axis of  $r(\tau,l)$  ( $L_{high\_peak} < \tau < t$ ) at the lag
         $L_{high\_peak}$ 
        Set threshold,  $Th_{Lag}$ , based on discriminant criterion measure
        Search smoothed  $r(\tau,L_{high\_peak})$  that is above  $Th_{Lag}$ ,
    End

    Group line segments that have almost the same section into a group
}
```

Figure 2.5. Pseudo code depicts the chorus detection procedure by Goto's RefraiD method.

2.2.4.2 Dynamic Programming

Dynamic programming is a very powerful algorithm paradigm in which a problem is solved by identifying a collection of subproblems and tackling them one by one, smallest first, using the answers to small problems to help figure out larger one, until the whole lot of them is solved [Dasgupta06]. Dynamic programming is another various approaches used to discover musical structure for later music thumbnailing [Maddage04] [Maddage06] [Chai(03b,03d,05)]. Chai [Chai(03b,03d,05)] uses dynamic programming to perform music pattern matching for finding repetitions in music and later discovering the structure of music. The structural analysis results determine actual alignment at section transitions, which is also similar to music segmentation. After the short-term segmentation process, the author computes the distance, c , between each two feature vectors. The computed distances are kept for later usage in a matrix scoring scheme. Two distances have been defined according to different dimensionalities of the used features. The distance between two one-dimensional pitch features v_1 and v_2 is defined as

$$d_p(v_1, v_2) = \frac{|v_1 - v_2|}{\text{normalization factor}} \quad (2.16)$$

The distance between two multi-dimensional feature vectors (FFT or chroma) \vec{v}_1 and \vec{v}_2 is defined as

$$d_f(\vec{v}_1, \vec{v}_2) = 0.5 - 0.5 \cdot \frac{|\vec{v}_1 \cdot \vec{v}_2|}{\|\vec{v}_1\| \|\vec{v}_2\|} \quad (2.17)$$

In both cases, the distance ranges between 0 and 1.

The computed feature vector sequence, $V[1, n] = \{v_j \mid j = 1, 2, \dots, n\}$, is segmented into segments of fixed length l . Each segment (i.e., $s_i = V[j, j + l - 1]$) is then matched with the feature vector sequence starting from this segment (i.e., $V[j, n]$) by using dynamic programming. The author first creates a $(n+1)$ -by- $(l+1)$ scoring matrix M_i , (as shown in Figure 2.6.) and then fills up the matrix based on a scoring scheme shown in Equation 2.17.

$$M(p, q) = \min \begin{cases} M[p-1, q] + e & (i \geq 1) \\ M[p, q-1] + e & (j \geq 1) \\ M[p-1, q-1] + c & (i, j \geq 1) \\ 0 & \text{otherwise} \end{cases} \quad (2.18)$$

where \mathcal{C} is the insertion or deletion cost, and \mathcal{D} is the distance between the two corresponding feature vectors mentioned above.

		V_j	$V_{(j+1)}$	$V_{(j+2)}$	$V_{(j+3)}$	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	V_n
	0	0	0	0	0	0
V_j	e											
$V_{(j+1)}$	2e											
...	...											
...	...											
...	...											
$V_{(j+l-1)}$	le											

Figure 2.6. Dynamic Programming Scoring matrix, M_i .

After the matrix fill step, the author performs a traceback step to determine the actual matching alignments that result in the minimum score. A repetition detection process is then performed by finding the local minima of the traceback results, $d_i[r]$, based on a predefined parameter h . The algorithm then merges consecutive segments that have the same repetitive properties into sections and generates pairs of similar section in terms of tuples $\langle j_1, j_2, shift \rangle$, which indicates the starting and ending location of each segment together with the lag information of its repetition. With the summarized repetition information, the music structure is inferred and labeled based on heuristic rules. Finally, the structure of music is revealed together with its section boundaries. With the use of structural analysis results, Chai summarizes the thumbnails of music by choosing the beginning or the end of the most repeated section based on criteria proposed by Logan and Chu [Logan00].

2.2.4.3 Clustering

Clustering is a grouping technique that has been extensively used in image processing, machine learning and data mining [Jain99]. Clustering organizes a set of objects into groups, such that all objects inside each group are somehow similar to each other. There exists an overwhelming amount of different clustering algorithms and criteria to determine the intrinsic grouping in a collection of data, and their selection depends on strategic factors. Logan and Chu [Logan00] use a clustering technique to discover the key phrase of music. The authors divide the sequence of features for the whole song into fixed-length contiguous segments, as a starting point. Then an iterative algorithm proceeds according the following steps:

1. Compute mean and covariance for each cluster with the assumption that each cluster has a Gaussian distribution.
2. Compute and store the distortion between each pair of cluster using a modified Kullback-Leibler distance measure [Siegler97]. The purpose of using Kullback-Leibler distance measure is to determine how close the two probability distributions are.
3. Select the pair of clusters with the lowest distortion between them.
4. If it is below a predefined threshold, combine these two clusters and go to step 1, else continue with step 5.
5. Each distinct cluster is assigned a label (such as ‘0’ and ‘1’), with all the frames inside this clusters are given this label.
6. Determine the most frequent label that occurs in the song.

By using this approach, Logan and Chu select the longest section (which consists of the most frequent label that appears in the first half of the song) as the key phrase of the song. Results from their evaluation test show that the clustering approach performed the best when compare to Hidden Markov Modeling and random selection. Nevertheless, the selected key phrase through clustering approach contains an unnatural starting and ending point, which is due to a limited resolution in the segmentation process.

Other than using K-means, a clustering technique that classify a given data set through a certain number of clusters (assume k clusters) fixed a priori, Foote and Cooper [Foote03] propose using Singular Value Decomposition (SVD). SVD is a dimension-reduction technique extensively used for still image segmentation, which can also be used for completing the task of segment clustering. SVD works by performing decomposition on a similarity matrix. In other words, it finds the repeated or substantially similar groups of segments through factoring a segment-indexed similarity matrix.

A few recent research works by [Abdallah05] [Levy06a] [Levy06b] propose their work of extracting classified structural segments, such as *intro*, *version*, *chorus*, *break and so forth*, from recoded music using a two atomization-clustering-agglomeration approach. The authors compute a sequence of short term HMM states occupancy histograms through the following sequence: (1) normalizing the extracted constant-Q log-power spectrum according to the estimated music beat length by means of beat tracking algorithm [Davies05]; (2) reducing feature vectors dimensionality using Principle Component Analysis (PCA); (3) a single Gaussian HMM is then fitted to the sequence of PCA coefficients and the Viterbi algorithm [Viterbi67] is used to decode the most probable state path which gives the most likely sequence of assignments for each beat of the music to the possible timbre-types. It is then followed by creating a sequence of short-term states occupancy histograms

over a sliding window of length w . The state histograms represent a distribution of the decoded timbre-types. Finally, a few clustering techniques such as Pairwise clustering [Hofmann97], with the used of the Kullback-Leibler divergence for defining the empirical dissimilarity measures between observed window states histograms, histogram clustering [Puzicha99] and K-means clustering are used separately to cluster the histograms into M clusters correspond to the segment type. So far, there exists no comparison with regard to which clustering methods perform better in discovering musical structure.

2.2.4.4 Hidden Markov Modeling

Hidden Markov Modeling (HMM) [Rabiner89] is another approach used in determining representative excerpts of music signal. HMM has a good capability in grasping the temporal statistical property of stochastic process. A Hidden Markov Model consists of a set of n finite number of states interconnected through probabilistic transitions, and is completely defined by the triplet, $\lambda = \{A, B, \pi\}$, where A is the state transition probability. B is the state observation probability, and π is the initial state distribution. At each time, HMM stays in one specific state. The state at time t is directly influenced by the state at time $t-1$. After each translation from one state to another, an output observation is generated based on an observation probability distribution associated with the current states. State variable are “hidden” and are not directly observable and thus, only output is observable. Figure 2.7 below shows a 4-state ergodic hidden Markov model.

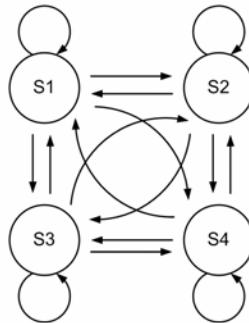


Figure 2.7. A 4-state ergodic hidden Markov Model

With the application of HMM, the segmentation process and the identification process are integrated into a single unified process [Logan00]. Hence, it is not necessary to perform any segmentation prior from using the HMM technique as the system learns the segmentation from the data itself. Unsupervised Baum-Welch is used to train the HMM given the sequence of feature attributes of the songs. In HMM, each state corresponds to a group of similar frames in the song. With Viterbi decoding, the most likely state sequence for the sequence of frames is determined, where each state is given a label. Finally, continuous segments are constructed by concatenating consecutive

frames that have the same given label. Logan and Chu [Logan00] chose the key phrase based on the duration and frequency of occurrence of these segments. In their studies, HMM overcame the problem of unnatural key phrase beginning that was observed using the clustering approach, even though the HMM did not achieve a satisfactory performance in the evaluation test. Nevertheless, using a fixed number of states in HMMs may not be an optimal solution since in real world music because the number of sections in music may vary significantly from one title to another one.

One underlying issue when using HMM in music structural analysis lies in finding the appropriate number of states for initialization. An insufficient number of states results in poor representations of the data, whilst an excessive amount of states causes too detailed representations. Besides, using a fixed preset number of states for the HMM model would also limit its potential in structure discovery. Hence, previous knowledge of these parameters will definitely improve the overall performance. Considering this factor, Peeters et al [Peeters02] propose a multi-pass approach combining segmentation and HMM, which does not require the a priori fixing of the number of states, for automatic dynamic generation of audio summaries. Its first-pass performs a long-term segmentation through similarity measurements between feature vectors in order to allow the definition of a set of templates (classes) of music. Here, the author intends to make use of the restructured information boundaries from long-term segmentation for achieving a better estimation of the number of classes and their potential states for a K-means clustering algorithm [MacQueen67]. With the constituted templates of the music, the second-pass organizes nearly identical (similarity ≥ 0.99) templates into groups and uses the reduced number of groups as “initial” states to initialize K-means clustering algorithm. The output from the K-means clustering is then used to initialize an ergodic HMM learning model, where every state of the model could be reached (in a single step) from every other state in a finite number of steps [Rabiner89]. Similar to Logan and Chu’s [Logan00] approach, the classic Baum-Welch algorithm is used to train the model. The outputs of the training are the state observation probabilities, the state transition probabilities and the initial state distribution. Finally, decoding using Viterbi algorithm with the given HMM and the signal features vectors, they obtain the state sequence corresponding to the piece of music. Through this unsupervised learning process, each time frame is given a state number. The authors suggest that the generation of the audio summary from this state representation can be done in several ways (with a given structure example: AABABCAB):

- Providing audio example of class transition ($A \rightarrow B$, $B \rightarrow A$, $B \rightarrow C$, $C \rightarrow A$)
- Providing a unique audio example of each of the states (A , B , C)
- Reproducing the class successions by providing an audio example for each class apparition (A , B , A , B , C , A , B)

- Providing only an audio example of the most important class in terms of global length or repetitiveness of the class) (A)

The research done by Peeters et al. has shown that an integrative approach by means of segmentation and an unsupervised learning method (K-means and Hidden Markov Models) can overcome the quick state-jump between states and produce a better and smoother state sequence. Thus, it improves overall the performance of using HMM in music structural discovery. However the authors do not provide any evaluation data to verify this observation.

2.3. Discussion

In this section, we discuss the pros and cons on each approach used in identifying representative musical excerpts of audio music. Self-similarity analysis approach has the advantage of providing a clear and intelligible view of audio structure. Nevertheless, it is not efficient for spotting repetitions with a certain degree of tempo change. A fixed resolution in its feature representation may give a different representation view on the tempo-changed repeated sections compared with its original section. Another problem with this approach is its threshold dependency in reducing noise for line segment detection. Threshold setting may vary from one song to another. Hence, a general setting threshold may not be valid for a wide range of audio.

The dynamic programming approach [Chai03c] has an advantage in offering a better accuracy in boundary detection. However this approach requires the comparison of all possible alignments between two sequences. The number of operations grows dramatically with the total number of frames. Thus, it suffers a lack of scalability.

The clustering approach manages to overcome the problem of the sensitivity to tempo changes suffered by the previously mentioned approaches, as long as boundary truncations are appropriate. However, one has to take notice that clustering, which organizes objects into groups based on their similarity, may produce complex representations of audio structure when a large range of similarity values exists among its feature contents. Hence, this approach is not appropriate for music that has non-homogenous feature contents, such as electronic music. Another limitation of this approach is that it is incapable in detecting unitary events in music. In other words, successive similar segments appear in music, for example a *verse* directly followed by another *verse*, will not be detected due to the homogeneity in the local properties of the signal. Only those with a distinct segment in between (e.g. *verse*→*refrain*→*verse*) will be detected instead.

HMM approach with its transition statistical parameters is capable of handling the problem caused by non-homogenous segments that we have to face with music content analysis. Other advantages of HMMs are their efficiency in handling non-fixed length input and their independency in completing both the segmentation task and identification task without any external support. Nevertheless, this approach has a disadvantage in its expensive computation. In addition, HMM's performance efficiency highly depends on the number of states and on a good initialization. An insufficient number of states causes a poor representation of the data, whilst excessive state numbers cause too detailed representations. As the number of states in HMM can roughly correspond to the amount of different sections in the song, using a fixed number of states in HMMs may generate unsatisfactory outcomes.

So far, much research works focusing on finding significant excerpts to represent a piece of music mainly depend on the repetitiveness of a music segment in the identification task. Apparently, no other assumptions have been proposed. In fact, how does one define the “significant” of an audio excerpt? From the musical point of view, it could be a “chorus” section of the pop music. While from the perceptual or cognitive point of view, it could be the most outstanding or attention grabbing or “strange” or “unexpected” excerpts that are usually not repeated but are capable in leaving a strong impression on our mind. Thus, repetitiveness may not be the only factor in criteria of defining the “significant” of an audio excerpt.

From music description extraction aspects, so far there are only a few research works [Lu04] [Chai05] reach the level where a unity and high-level of semantic description of music can be directly extracted from the music signals. However the related research works have not addressed the problem of modulation within song even though it is one of the most common phenomena encountered in music structural analysis task. Goto's [Goto03a] *RefrainD* system has addressed the modulation within song issue, however the system so far only reaches the level of detecting specific music sections with the remaining sections left without being identified. Whereas from the application aspects, current significance of discovering the structural description of music seems to only point towards audio browsing and music thumbnailing or summarization contexts. Prior knowledge of such a high-level description of music should be able to give a better grasp of the musical data and further improve the content analysis and processing of the acoustics signals. However so far there exists no exploration with regard to the practical usability of music structural descriptions in other contexts besides the above mentioned area.

Finally, by reviewing the current developments in this area, we observed a few limitations in the aspect of algorithm evaluations in present literature. First limitation is the lack of generality of the test

databases. Hence, by using such a database, it is quite impossible to obtain an objective evaluation on the algorithm efficiency for most of the existing music. Another limitation is the method in weighting the importance of extracted music sections. The significance of the musical excerpts in audio signal highly depends on human perception.

2.4. Summary

In this chapter, we have covered a substantial range of background information in music structural analysis. We have presented various audio feature classes and extraction approach used for music structural analysis, audio segmentation techniques for better truncations of audio signal and related identification approaches for discovering the structure of music.

In the next chapter, we begin our study of segmenting audio semantically in term of the structural changes in the music signal. Chapter 3 begins with a presentation of the overview of our proposed framework for semantic audio segmentation. The chapter proceeds by presenting the descriptions of our proposed approach in more detail. Finally, we present quantitative evaluation results of the performance of our proposed method using our own test database.

Chapter 3

Semantic Audio Segmentation

In this chapter, we address the problem of finding acceptable structural boundaries, without prior knowledge of music structure. This provides a way to separate the different “sections” of a piece, such as “intro”, “verse”, “chorus”, etc. From now on, we use semantic audio segmentation to refer to our proposed method in this respect. There is a second related problem consisting of assigning labels to the segments that are found. Here, we will not go into this issue but leave it to the following chapter instead.

The work in this chapter has two aspects. First, we propose our semantic audio segmentation method. In our approach, we divide the segment boundaries detection task into a two-phase process with each having different functionalities. Unlike traditional audio segmentation approaches, we employ image processing techniques to enhance the significant segment boundaries in audio signals. In order to obtain appropriate structural boundaries, we propose a combination of low-level descriptors to be extracted from the music audio signal. Section 3.1 comprises complete descriptions of this aspect. First, we start by providing an overview of the proposed framework. Later, we extend the description of each procedure in detail at each subsection, according to the processing sequence.

The second aspect, presented in Section 3.2, is a set of experiments to evaluate the efficiency of our system by means of using various combinations of low-level descriptors and descriptive statistics. We use some basic measures in evaluating search strategies to achieve an objective evaluation for each performed experiment. Two different datasets are used to evaluate the performance of our algorithm. The first dataset consists of 54 songs from the first four CD’s of The Beatles’ (1962 - 1965), whereas the second dataset comprises of 27 pop songs from the Magnatune database. The

experiment results evaluated using both test datasets show that our approach achieves an overall effectiveness as high as 74% in identifying structural boundaries in music audio signals.

3.1. Approach

Audio segmentation facilitates partitioning audio streams into short regions. It seems an indispensable process in certain content-based applications, such as audio notation, audio summarization, and audio content analysis. Due to this reason, research in this area has received increasing attention in recent years. A number of different approaches have been proposed [Aucouturier01, Foote00, Tzanetakis99, Ong04].

In this chapter, we propose a novel approach for the detection of structural changes in audio signals by dividing the segment detection process into two phases (see Figure 3.1). Each phase is given a different goal: Phase 1 focuses on detecting boundaries, which may contain structural changes from the audio signal; Phase 2 focuses on refining detected boundaries obtained from phase 1 by aggregating contiguous segments while keeping those which mark true structural changes in the music audio. Our proposed method consists of 9 steps as follows:

Phase 1 – Rough Segmentation

- (1) Segment input signal into overlapped frames of fixed length and compute audio descriptors for each frame (see section 3.1.1);
- (2) Compute between-frames cosine distance to obtain several similarity matrices [Foote00] for each one of the used features (see section 3.1.2);
- (3) Apply morphological filter operations (see section 3.1.2) to similarity matrices for enhancing the intelligibility of the visualization;
- (4) Compute novelty measures by applying kernel correlation [Foote00] along the diagonal of the post-processed similarity matrices (see section 3.1.2);
- (5) Detect segments by finding the first 40 highest local maxima from novelty measure plot (see section 3.1.2);
- (6) Combine the detected peaks to yield boundary candidates of segment changes of music audio (see section 3.1.2);

Phase 2 – Segment Boundaries Refinement

- (7) Assign frames according to detected segments obtained from phase 1 and compute the average for all the used features (see Table 3.1) in each segment;
- (8) Compute between-segments distances using the mean value of each feature in each segment (see Table 3.1);

- (9) Select significant segments based on distance metrics (see Table 3.1).

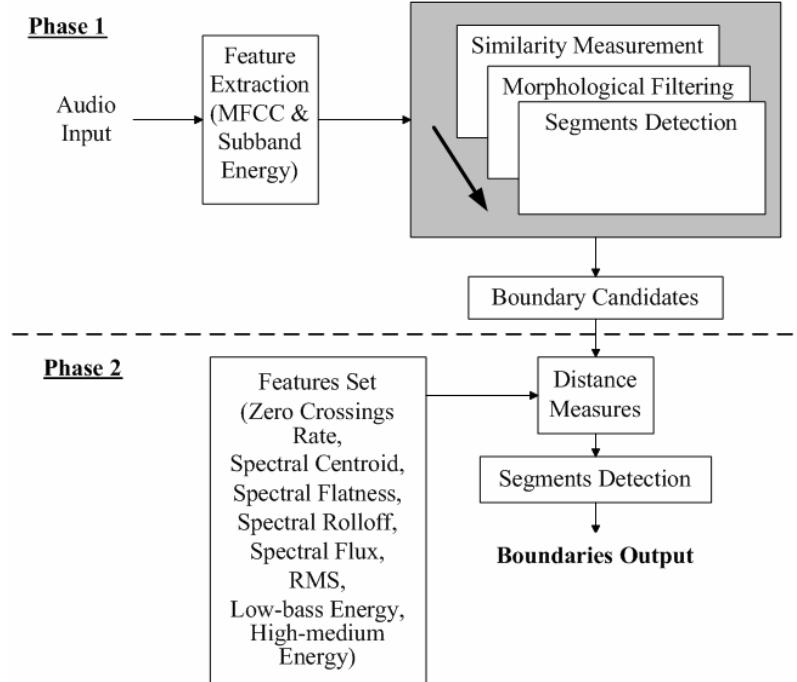


Figure 3.1. Overview framework of our approach.

The following sections explain each step in detail.

3.1.1. Feature Extraction

We begin our segmentation task by extracting a feature representation of the audio content. As mentioned earlier, detecting significant structural changes in music signals is a key issue of our research objective for this chapter. Thus proper selection of feature attributes is crucial to obtain appropriate musical content descriptors that grant a proper boundaries-detection process. Nevertheless, an effective description of musical content not only depends on the best feature attributes, but sometimes also depends on the use of different features in a combined manner. Therefore, the application of musical knowledge into the selection process would further improve the quality of musical content description.

With regards to obtaining the short-term descriptions of the audio sound signal, we partition the input signal into overlapped frames (4096-samples window length) with a hop size of 512 samples. We then follow by extracting feature descriptions of each of these frames with a Hamming window.

To estimate the content descriptions of the music audio signal, we consider different timbre and dynamics related features: MFCC, sub-bands energy, spectral centroid, spectral rolloff, spectral flux, zero crossings, spectral flatness, low bass energy, high-medium energy and RMS energy. The following gives a brief description of each of the used content descriptors. Please refer to section 2.2.1 for detailed explanations of these descriptors.

MFCC, also called Mel-Frequency Cepstral Coefficients: A compact representation of an audio spectrum that takes into account the non-linear human perception of pitch, as described by Mel-scale [Rabiner93].

Sub-bands energy: A measure of power spectrum in each sub-band. We divide the power spectrum into 9 non-overlapping frequency bands as described in [Maddage04].

Spectral Centroid: A representation of the balancing point of the spectral power distribution within a frame.

Spectral Rolloff: A measure of frequency, which is below the 95th percentile of the power spectral distribution. It is a measure of “skewedness” of the spectrum.

Spectral Flux: The 2-norm of the frame-to-frame spectral magnitude difference vector. It measures spectral difference, thus it characterizes the shape changes of the spectrum.

Zero Crossings: A time-domain measure that gives an approximation of the signal’s noisiness.

Spectral Flatness: A measure of the flatness properties of the spectrum within a number of frequency bands. High deviation from a flat shape might indicate the presence of tonal components.

High-medium energy: A ratio of the spectrum content within the frequency range of 1.6 kHz and 4 kHz to the total content. This frequency range comprises all the important harmonics, especially for sung music.

Low-bass energy: A ratio of the low frequency component (up to 90 Hz) to the total spectrum energy content. This frequency range includes the greatest perceptible changes in “bass responses”.

RMS energy: A measure of loudness of the sound in a frame.

In our approach, we use two different groups of descriptors, one for each of the different phases. In Phase 1, descriptors are used to detect segment boundaries, which hold a significant timbre change between its previous and next compared frames. In the case of phase 2, the descriptors are mainly used to refine the detected segment boundaries from phase 1.

Phase 1	Phase 2
MFCC Sub-bands Energy	Zero Crossings rate, Spectral Centroid, Spectral Flatness, Spectral Rolloff, Spectral Flux, RMS, Low-bass Energy, High-medium Energy

Table 3.1. The list of audio descriptors for Phase 1 and Phase 2.

3.1.2. Phase 1 – Rough Segmentation

After computing feature vectors for each frame, we group every 10 frames (116ms) and calculate the mean value for every feature. In this phase of the segment detection process, we only work with multidimensional features (i.e. MFCC and sub-band energies). We treat those features separately in order to combine both results in the final stage of the detection process in phase 1. In order to find the structural changes in the audio data, we measure the distance between each feature vector, $V_n = \{v_{n,1}, v_{n,2}, \dots, v_{n,m}\}$, and its neighbouring vectors, $V_{n+i} = \{v_{n+i,1}, v_{n+i,2}, \dots, v_{n+i,m}\}$, using cosine angle distance [Foote00] given by the expression:

$$SD_{\text{cosine}}(V_n, V_{n+i}) = \frac{V_n \cdot V_{n+i}}{\|V_n\| \|V_{n+i}\|} \quad (3.1)$$

$$= \frac{\sum_{j=1}^m (v_{n,j} \cdot v_{n+i,j})}{\sqrt{\sum_{j=1}^m (v_{n,j})^2 \cdot \sum_{j=1}^m (v_{n+i,j})^2}} \quad (3.2)$$

where m denotes the m -dimensional of the feature vector.

Figure 3.2 illustrates the two-dimensional cosine similarity plot computed using MFCC features. As shown in the figure, some structural changes can be perceived in the similarity plot. To enhance

such information, we need to further improve the intelligibility of the vague visualization given by the similarity plot. For this purpose, we apply morphological filters [Burgeth04], a widely used filtering technique applied to image processing, on the computed distance matrix representations. The reason for selecting the morphological filter lies in its advantage in preserving edge information and its computational efficiency over other techniques. Different from Lu's previous work in structural analysis [Lu04], the idea behind using morphological filtering operations here is to increase the intelligibility of the structural changes and facilitate the enhancement of the segment boundaries instead of removing redundant short lines from the time-lag matrix. Since morphological filtering techniques are relatively unknown in music analysis, we dedicate a few paragraphs below to providing a brief introduction about its operations' functionalities and implementations procedure.

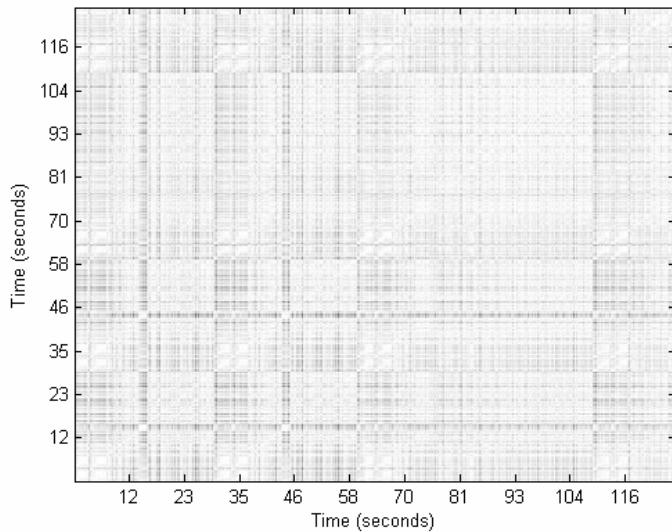


Figure 3.2. Two-dimensional cosine similarity plot computed from the song entitled *When I Get Home* using MFCC features.

Morphology filtering is an analysis process of signal in terms of shape. Basically, it uses set theory as the foundation for many of its operations [Young02]. Its simplest operations are dilation and erosion. In general, dilation causes objects to dilate or grow in size while erosion causes objects to shrink. The amount of changes (growth or shrinkage) depends on the choice of the structuring element. The following paragraph explains how dilation and erosion work in detail. Dilation, also known as 'Minkowski Addition', works by moving the structuring element over the input signal and the intersection of the structuring element reflected and translated with the input signal is found. In another words, the output is set to one unless the input is the inverse of the structuring element. For instance, '000' would cause the output to be zero and placed at the origin of the structuring element, B, for the given example in figure 3.3.a. Similar to dilation, erosion, also known as 'Minkowski

'Subtraction', works by moving the structuring element over the input signal. The erosion of the input signal, A, and the structure element, B, is the set of points x such that B translated by x is contained in A. In contrast with the dilation operation, the output is set to zero unless the input is identical to the structuring element. Figure 3.3.b shows how erosion opens up the zeros and removes runs of ones that are shorter than the structuring element in a one-dimensional binary signal. [Young02].

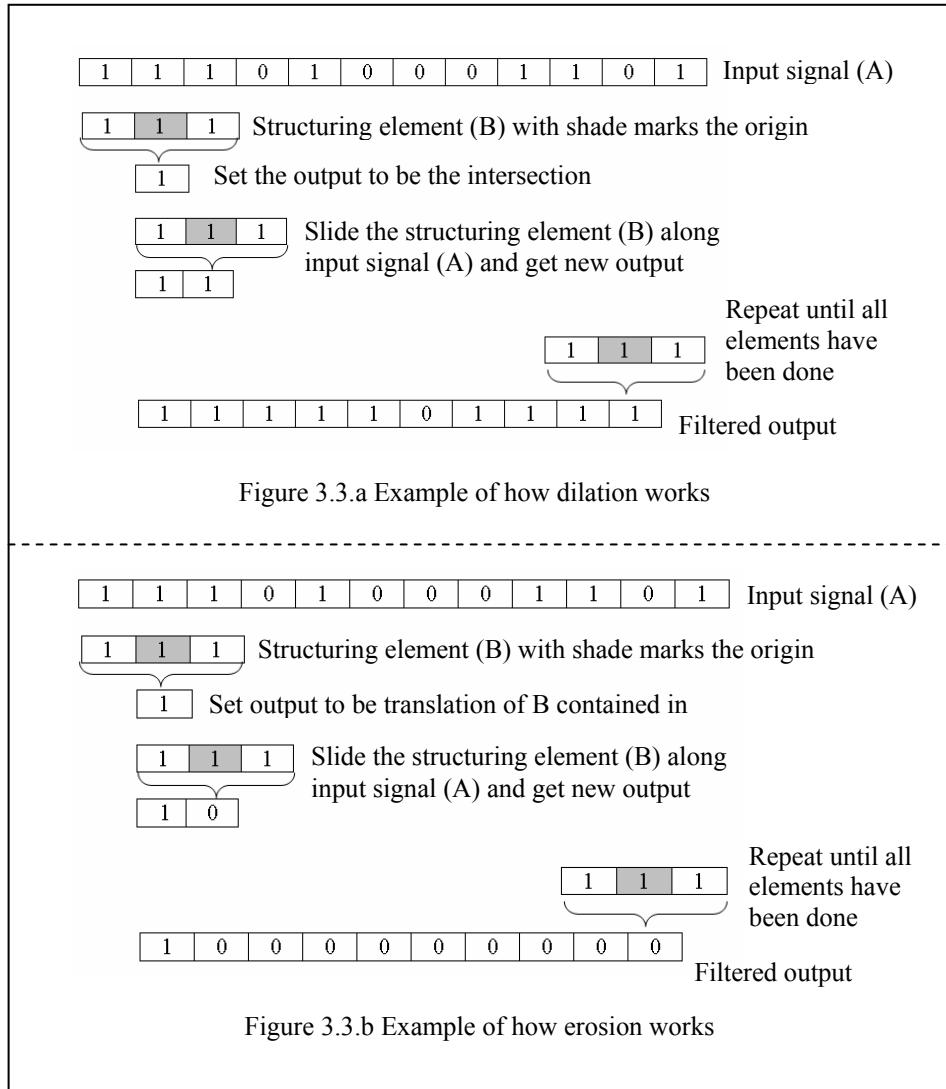


Figure 3.3. Examples of how dilation and erosion work with the shaded structuring elements show the origin element.

So far, the above mentioned dilation and erosion operations are associated with one-dimensional binary signals. For non-binary signals, the dilation (erosion) operation works the same as taking the

maximum (minimum) value of the signal, which lies within the 1's of the structuring element. Thus, dilation and erosion operations for non-binary signals can be redefined as

$$\text{Dilation} = \max_{i \in B}(A_x) \quad \text{where, } |i| \leq \frac{n-1}{2} \cap i \in \mathbb{Z} \quad (3.3)$$

$$\text{Erosion} = \min_{i \in B}(A_x) \quad \text{where, } |i| \leq \frac{n-1}{2} \cap i \in \mathbb{Z} \quad (3.4)$$

Figure 3.4 illustrates the properties of the input signal, A_x , with its structuring element, B_i , as defined in expressions 3.3 and 3.4.

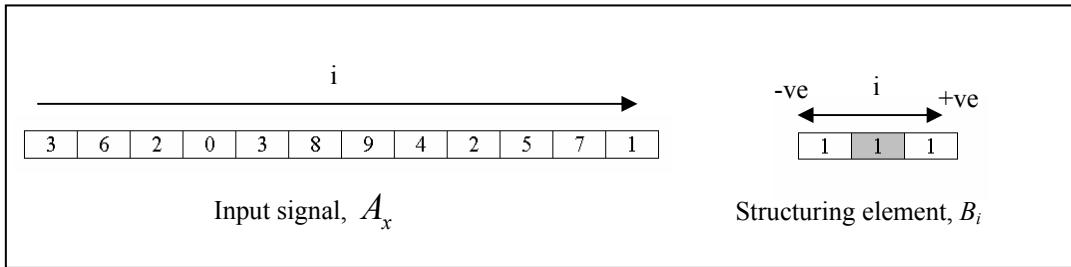


Figure 3.4. The properties of one-dimensional signal, A_x , with its structuring element, B_i , in defined in expressions 3.3 and 3.4.

For two-dimensional input signals, erosion and dilation operations still work in exactly the same way as for one-dimensional input signals, but with a two-dimensional structuring element instead. Hence, dilation and erosion operations for two-dimensional signals can be expressed as:

$$\text{Dilation} = \max_{(i,j) \in B}(A_{x+i,y+j}), \quad \text{where } |i|, |j| \leq \frac{n-1}{2} \cap i, j \in \mathbb{Z} \quad (3.5)$$

$$\text{Erosion} = \min_{(i,j) \in B}(A_{x+i,y+j}), \quad \text{where } |i|, |j| \leq \frac{n-1}{2} \cap i, j \in \mathbb{Z} \quad (3.6)$$

Figure 3.5 shows the two-dimensional properties of an input signal, $A_{x,y}$, with its n -by- n structuring element, $B_{i,j}$, as defined in expression 3 and 4.

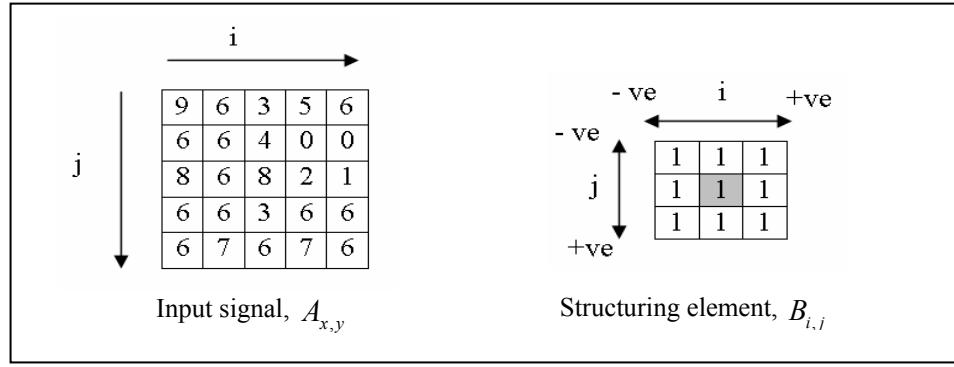


Figure 3.5. The two-dimensional properties of input signal, $A_{x,y}$, with its n -by- n structuring element, $B_{i,j}$, as defined in expression 3.5 and 3.6.

‘Opening’ and ‘Closing’ operations are two morphological filter operations, which contain the properties of both dilation and erosion operations. The ‘Closing’ operation works by dilating the signal and is followed by eroding the results. In contrast, the ‘Opening’ operation works by eroding the signal followed by dilating the results. Figure 3.6 demonstrates both ‘Closing’ and ‘Opening’ operations of the morphological filter on a one-dimensional binary signal. From the figure, we can clearly see the distinct properties of these two operations. The ‘Closing’ operation (as shown in Figure 3.6.a) closes the gaps that lie within the length of the structuring element, whereas the ‘Opening’ operation (as shown in Figure 3.6.b) opens the gaps and removes runs of ones that are shorter than the structuring element of the signal. Otherwise, the signal is left unchanged.

In our work, we utilize ‘Open-Close’ and ‘Close-Open’ operations of the morphological filter. These two operations are the combination products of ‘Opening’ and ‘Closing’ operations in order to merge their properties into one filter operation. The ‘Open-Close’ operation is implemented by first opening the signal and then closing the opened signal. In contrast with the ‘Open-Close’ operation, the ‘Close-Open’ operation is implemented by first closing the signal and then opening the closed signal. Both ‘Open-Close’ and ‘Close-Open’ can be expressed as:

$$\text{Open-Close}(A, B) = \text{close}(\text{open}(A, B), B) \quad (3.7)$$

$$\text{Close-Open}(A, B) = \text{open}(\text{close}(A, B), B) \quad (3.8)$$

where A denotes the input signal and B is the structuring element.

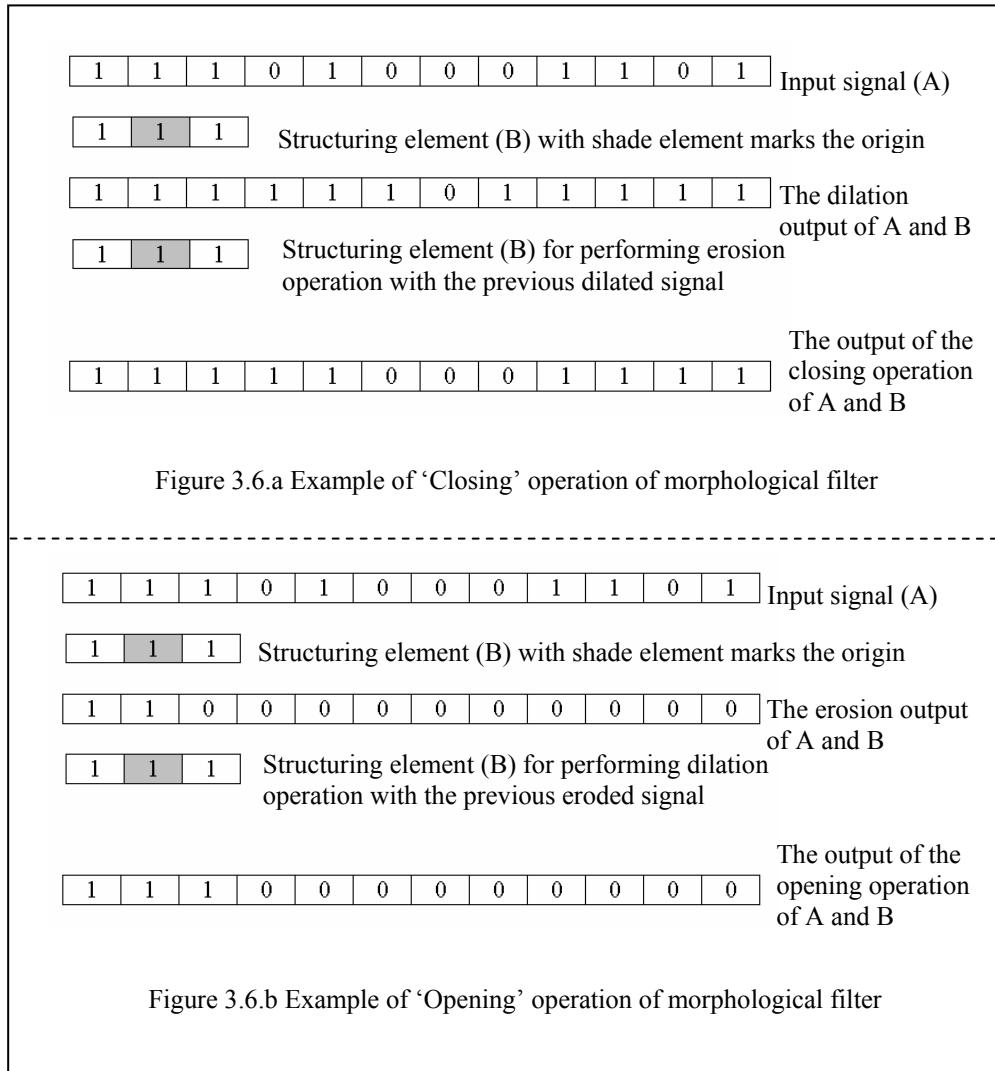


Figure 3.6. The opening operation of the morphological filter on a one-dimensional binary signal.

Figure 3.7 shows how these filter operations work on a one-dimensional binary signal. Comparing the operations outputs as shown in Figure 3.6 with those in Figure 3.7, we can see that ‘Open-Close’ (‘Close-Open’) operations produce a similar output as ‘Opening’ (‘Closing’) operations when dealing with one-dimensional binary signals. However this is not the case when applied to two-dimensional non-binary signals. ‘The Opening’ operation will remove high intensity points whilst keeping the rest of the signal intact. The ‘Closing’ operation will discard low valued points whilst keeping the rest of the signal intact. However, the ‘Open-Close’ and ‘Close-Open’ operations will remove both high and low valued points while keeping the rest of the signal intact.

<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td></tr></table>	1	1	1	0	1	0	0	0	1	1	0	1	Input signal (A)
1	1	1	0	1	0	0	0	1	1	0	1		
<table border="1"><tr><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	Structuring element (B) with shade element marks the origin									
1	1	1											
<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	1	1	0	0	0	0	0	0	0	0	0	The output of ‘Open-Close’ operation of A and B
1	1	1	0	0	0	0	0	0	0	0	0		
<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	1	0	0	0	1	1	1	1	The output of ‘Close-Open’ operation of A and B
1	1	1	1	1	0	0	0	1	1	1	1		

Figure 3.7. The ‘Open-Close’ and ‘Close-Open’ operations of the morphological filter on a one-dimensional binary signal.

Since our computed distance matrix consists of two-dimensional non-binary signals, the applications of ‘Open-Close’ and ‘Close-Open’ operations would disregard high and low valued points in our distance matrix and produce (dis)similarity representations with an enhanced intelligibility. Figure 3.8 and Figure 3.9 illustrate the post-processed distance matrices obtained from applying ‘Close-Open’ and ‘Open-Close’ operations independently on the distance matrix as shown in Figure 3.2. Compared to Figure 3.2, the appearances of structural patterns in the distance representation plots have been amplified after morphological filtering processes. From the figures, we can see that although both two operations have the same characteristics in removing intensity points from the signal, they do not produce the same filter results. This is due to the different sequence of erosions and dilations in implementing both operations. In addition to the outputs of both morphological operations, we also utilize additional distance matrices yielded from multiplication and subtraction between ‘Close-Open’ and ‘Open-Close’ operation outputs to facilitate the identification of relevant structural changes of music. Figure 3.10 illustrates the distance matrix representation obtained from the multiplication of ‘Open-Close’ and ‘Close-Open’ filter results.

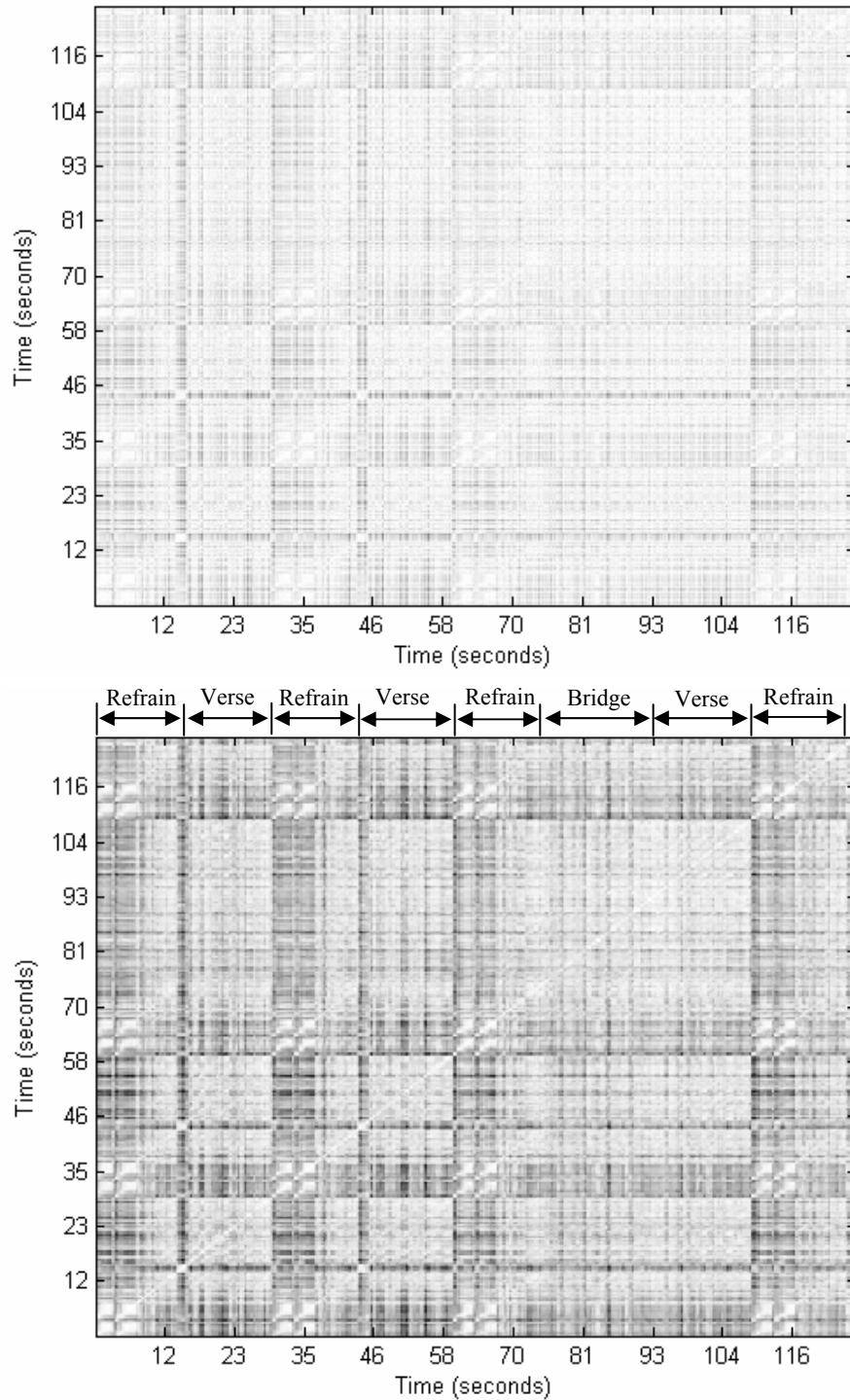


Figure 3.8. Similarity representation before morphological operation (top) versus similarity representation after ‘Close-Open’ operation.

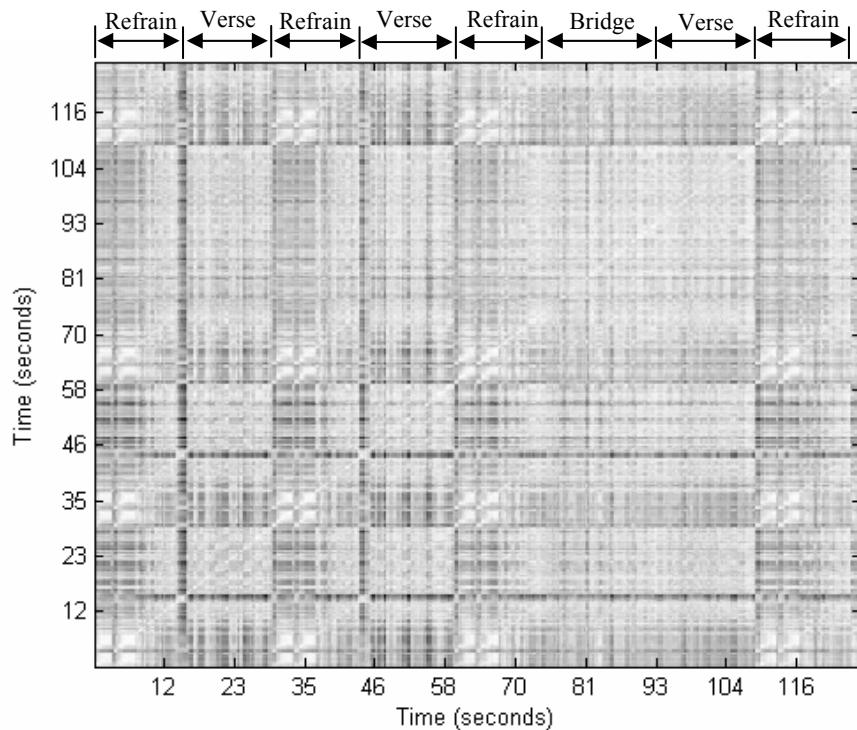


Figure 3.9. Similarity representation after ‘Open-Close’ operation

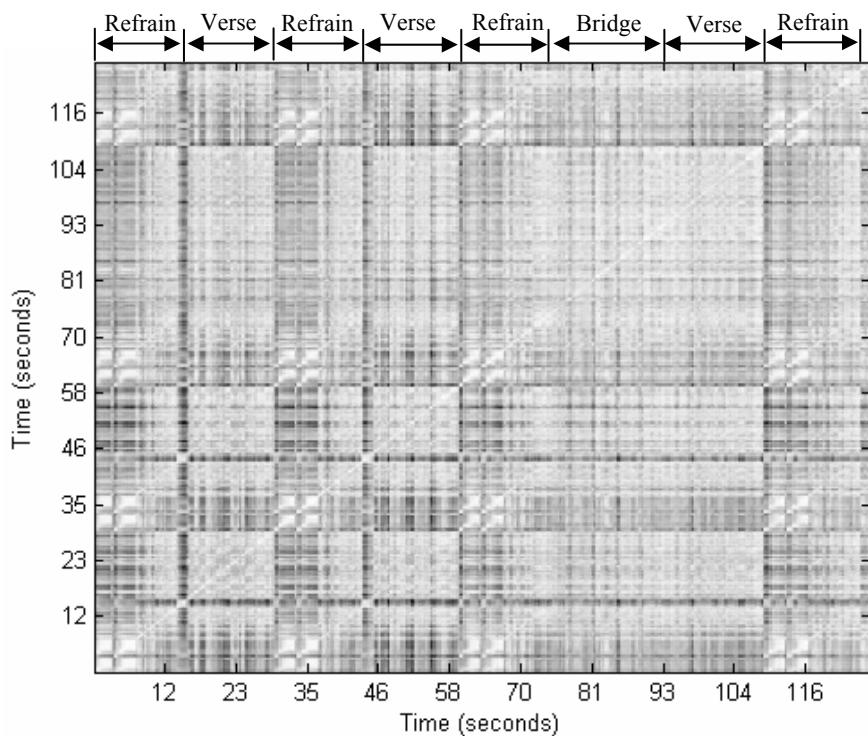


Figure 3.10. Distance matrix representation obtained from the multiplication between ‘Open-Close’ and ‘Close-Open’ filter results.

With the post-processed similarity matrices obtained from morphological filtering operations, we then apply a kernel correlation [Foote00], with a width of 10, along the diagonal of each post-processed similarity matrix to measure the audio novelty. This is to observe any significant changes of the related audio contents for approximately every 1 second. In order to accumulate all information from the post-processed similarity matrices, we aggregate all the computed novelty measures and normalize it with its maximum value. This produces an overall novelty measure with values within the range of 0 to 1. Based on the overall novelty measure, the first 40 highest local maxima are selected based on the constraint that each selected local maximum must be at least m seconds apart from its neighbouring selected local maximum. Three empirical preset m parameters were considered: 2.3 sec, 2.9 sec, and 3.5 sec. Finally, we accumulate all the peaks detected based on these three m parameters and select the first 40 highest local maxima amongst all local maxima as the segment boundaries candidates from the employed features.

As mentioned earlier, in this phase of the segment detection process, we are working with MFCC and sub-band energies. Hence the whole process of similarity measurement, morphological filtering and segment candidates' selection is repeated for sub-band energies. Finally, we combine all segment boundaries candidates detected from both features (MFCC and sub-band energy) and select the highest 40 amongst them to be considered as boundaries candidates of segment changes of music audio. The selection is based on the criterion that each segment candidate must be at least 2.3 seconds (the lowest considered value for m parameters) apart from each other. Figure 3.11 illustrates the detected boundaries candidates yielded by the segment detection process in phase 1.

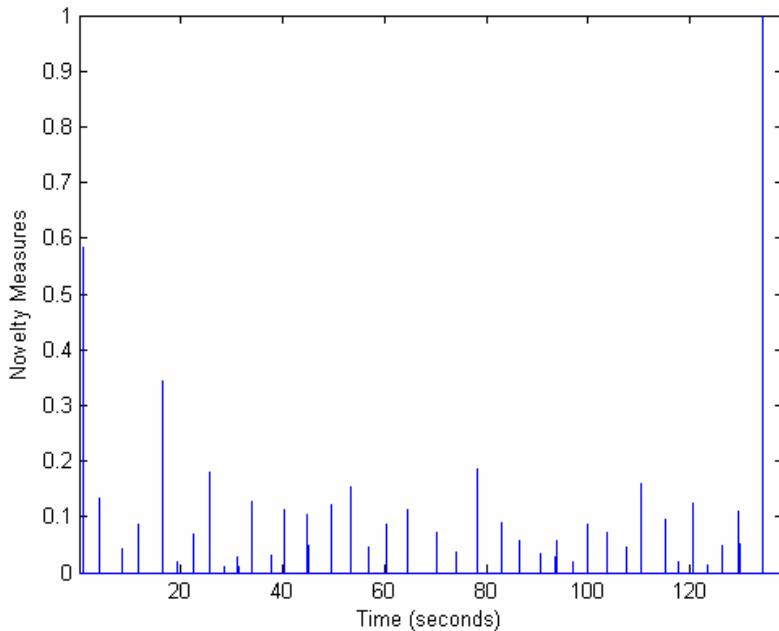


Figure 3.11. Candidate boundaries yielded by segment detection process in phase 1.

3.1.3. Phase 2 – Segment Boundaries Refinement

In the second phase of the detection process, we only use the time position from the candidates extracted in the previous phase. Here, we consider the values for all the attributes used in phase 2 (see Table 3.1) that are within the detected segment boundaries and compute the average of each. Hence, each detected segment now comprises only a set of feature vectors representing the mean value of the attributes in that segment. It has to be mentioned that in our attributes, there exists a different range of feature values. Presumably, attributes whose values are larger than others would have more influence in determining the similarity of any two sequences. Hence, in order to avoid such an effect and to have an equal importance weight among the used attributes, we normalize all attributes so that its feature values are within the ranges of 0 and 1. We then compute the (dis)similarity between each segment and its neighbouring segments by measuring the Euclidean distance between their feature vectors. The Euclidean distance between vectors $V_n = \{v_{n,1}, v_{n,2}, \dots, v_{n,m}\}$ and $V_{n+i} = \{v_{n+i,1}, v_{n+i,2}, \dots, v_{n+i,m}\}$ is given by the expression:

$$|V_n - V_{n+i}| = \sqrt{\sum_{j=1}^m (v_{n,j} - v_{n+i,j})^2} \quad (3.9)$$

where m denotes the m -dimensional of the feature vector. Theoretically, the Euclidean distance and cosine angle distance used in section 3.1.2 give the same distance measure when two compared feature vectors have same variance values [Gang02]. In fact, the cosine angle distance is very sensitive to the variance of compared feature vectors. Thus, it is very useful in finding very similar items. Since our feature vectors in this phase are obtained based on the detected boundaries information from phase 1, we hypothesize that the Euclidean distance should be more suitable to compute the distance measures of these feature vectors. Similar to the previous steps in computing novelty measures from the similarity representations, we apply a kernel correlation, along the diagonal of the (dis)similarity representation of segments to yield the novelty measures, N , between each segment and its next sequential segment. Figure 3.12 and Figure 3.13 illustrate the (dis)similarity representations and novelty measures computed from the (dis)similarity representations between segments. Finally we select significant segment boundaries from the computed novelty measures, $N = \{n_s \mid s = 1, 2, \dots, l\}$ (where l is the number of segment boundaries candidates) based on the following steps:

1. Select all the peaks that lie above a predefined threshold, P_t , based on their computed novelty measures, N_s , and organize them into a group, which is represented as $P = \{p_i \mid i = 1, 2, \dots, M\}$ (M is the number of selected peaks). Whereas those peaks that lie below the predefined threshold, P_t ,

are organized into another group denoted by $E = \{e_j \mid j = 1, 2, \dots, N\}$ (N is the number of unselected peaks).

2. Organize all peaks in E in ascending order according to their distance measures.
3. Select the highest peak in E for further evaluation.
4. Based on temporal information, if the evaluated peak is located at least 4 sec apart from any peaks in P , insert it in group P and reorganize all peaks in group P in ascending order based on the segment index number; otherwise delete it from E . This is based on the assumption that each section in music (e.g. verse, chorus, etc.) should at least hold for 4 sec (1 bar for songs with quadruple meter with 60 bpm tempo) in length before moving to the next section.
5. Go to step 3.

The whole iterative peak selection process ends when there are no more peaks in E . Finally, segment boundaries in P are considered as significant segment boundaries that mark structural changes in music audio signals.

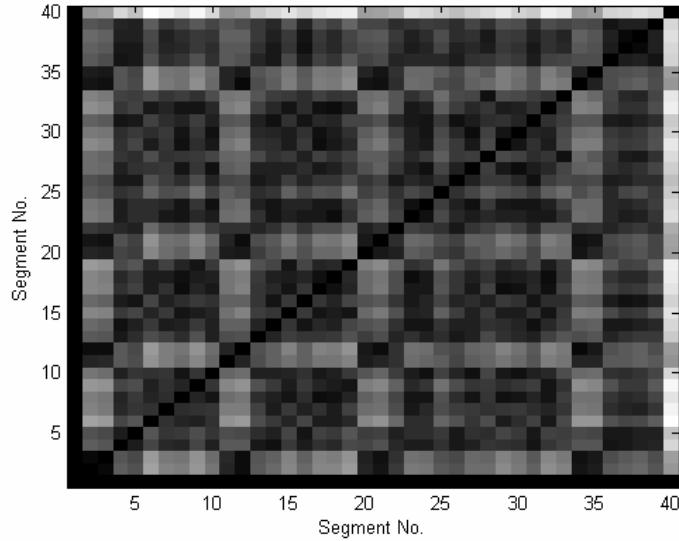


Figure 3.12. The (dis)similarity representations between segments detected in phase 1.

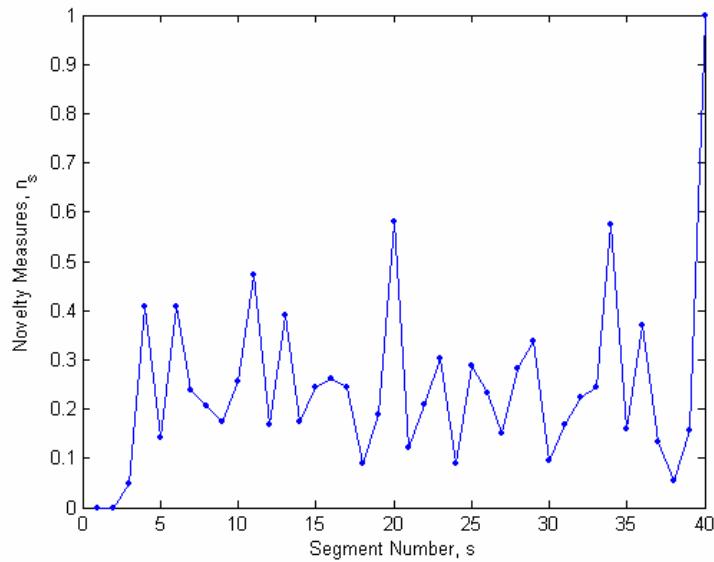


Figure 3.13. The novelty measures computed from the (dis)similarity representations between segments.

3.2. Evaluation

In this section we present an evaluation of our proposed semantic audio segmentation method in detecting structural changes in music audio signals. We first begin by discussing in detail our dataset. This is followed by presenting the measures used in evaluating the performance of our proposed method.

3.2.1. Datasets

In our experiment, we created an audio database, which consists of 54 songs from the first four CD's of The Beatles (1962 – 1965) as a test set. In addition, we also created another test set, containing 27 pop songs from the Magnatune⁵ database. The reason of using a second test set is to avoid having an evaluated performance result that biases towards The Beatles' songs. Each song in both test sets is sampled at 44.1 kHz, 16-bit mono. In the objective evaluation, we have generated a ground truth by manually labelling all the sections (i.e. intro, verse, chorus, bridge, verse, outro, etc.) of all the songs in the test sets. For The Beatles' music test set, the ground truth is manually labelled according to the information provided by Allan W. Pollack's "Notes On" Series website on song analyses of The Beatles' twelve recording projects⁶. Since there exists no official song analyses available for Magnatune's songs, we generated the ground truths by comparing labellings manually annotated by two advanced music conservatory students who listened to the music and generated the boundary marks. A music composer supervised the labelling process and results.

3.2.2. Procedure

To quantitatively evaluate the detected segments from the proposed algorithm, the detected segment boundaries are compared with the ground truth in term of precision and recall. The precision and recall are defined as follows:

$$\text{Precision} = \frac{D \cap G}{D} \quad (3.10)$$

$$\text{Recall} = \frac{D \cap G}{G} \quad (3.11)$$

where D denotes detected segments, G denotes relevant ground truth segments and $D \cap G$ signifies detected segments that are placed within the region of relevant ground truth segments' with its tolerance deviation. In evaluating the identified segments, we allow a tolerance deviation of ± 3 seconds (approximately 1 bar for a song of quadruple meter with 80 bpm in tempo) from the manually labelled boundaries.

⁵ Magnatune official web page: <http://magnatune.com/>

⁶ The Twelve Recording Projects of the Beatles web page: http://www.icce.rug.nl/~soundscapes/DATABASES/AWP/awp-beatles_projects.html

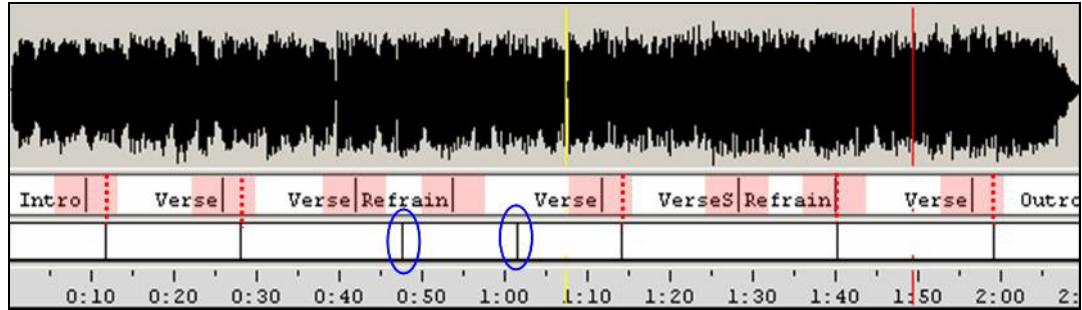


Figure 3.14. An example of measuring segmentation performance with a tolerance deviation presented as shaded area (top transcription: ground truth segments; bottom transcription: detected segment). Circled segments mark the outliers of the correctly detected segments.

Figure 3.14 shows an example of measuring segmentation performance with a tolerance deviation. In the example shown in the figure 3.14, segments with marked circles do not fall within the region of ground truth segment boundaries with its tolerance deviation (shaded area). Hence these two mark circled segments will not be considered as $D \cap G$. The top transcription, which represents the ground truth results, comprises 8 segment boundaries whereas the bottom transcription, which denotes detected results, comprises 7 segment boundaries. Thus, the precision and recall in this example are 5/7 (≈ 0.71) and 5/8 (≈ 0.63), respectively.

Precision and recall measures are mainly used to evaluate the accuracy and reliability of the proposed algorithm. In addition, we use the F-measure to evaluate overall effectiveness of segment detection by combine recall and precision with an equal weight:

$$\text{F-measure} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \quad (3.12)$$

3.2.3. Results and Discussion

In this section, we summarize the key findings from the quantitative evaluation results of our proposed segmentation method discussed in this chapter.

Bar graphs in Figure 3.15 and Figure 3.16 show the precision, recall and F-measure scores obtained for all songs in both our test databases with a tolerance deviation of ± 3 seconds. From the first test set, our proposed structural changes detection approach has achieved an average precision higher than 72% and an average recall of 79% using the ground truth set. The overall F-measure has reached 75%. In other words, with 10 detected segment boundaries, 7 of them are correctly detected

compared to the ground truth data. Whilst about 2 out of 10 manually labelled boundaries are missed by our automatic boundaries detector. The distribution of precision scores has a standard deviation of 0.11 and the range of precision values spans across 0.41-0.94. For the obtained results, we also observe that the best performance in The Beatles' test set is in the case of SongID-35 with its recall and precision score of 100% and 95%, respectively. Whereas the worst performance is observed in the case of SongID-47, which only reaches the recall rate of 38% and precision rate of 56%. Figure 3.17 illustrates the segment boundaries detected by our proposed algorithm, with manually labelled segment boundaries for SongID-35.

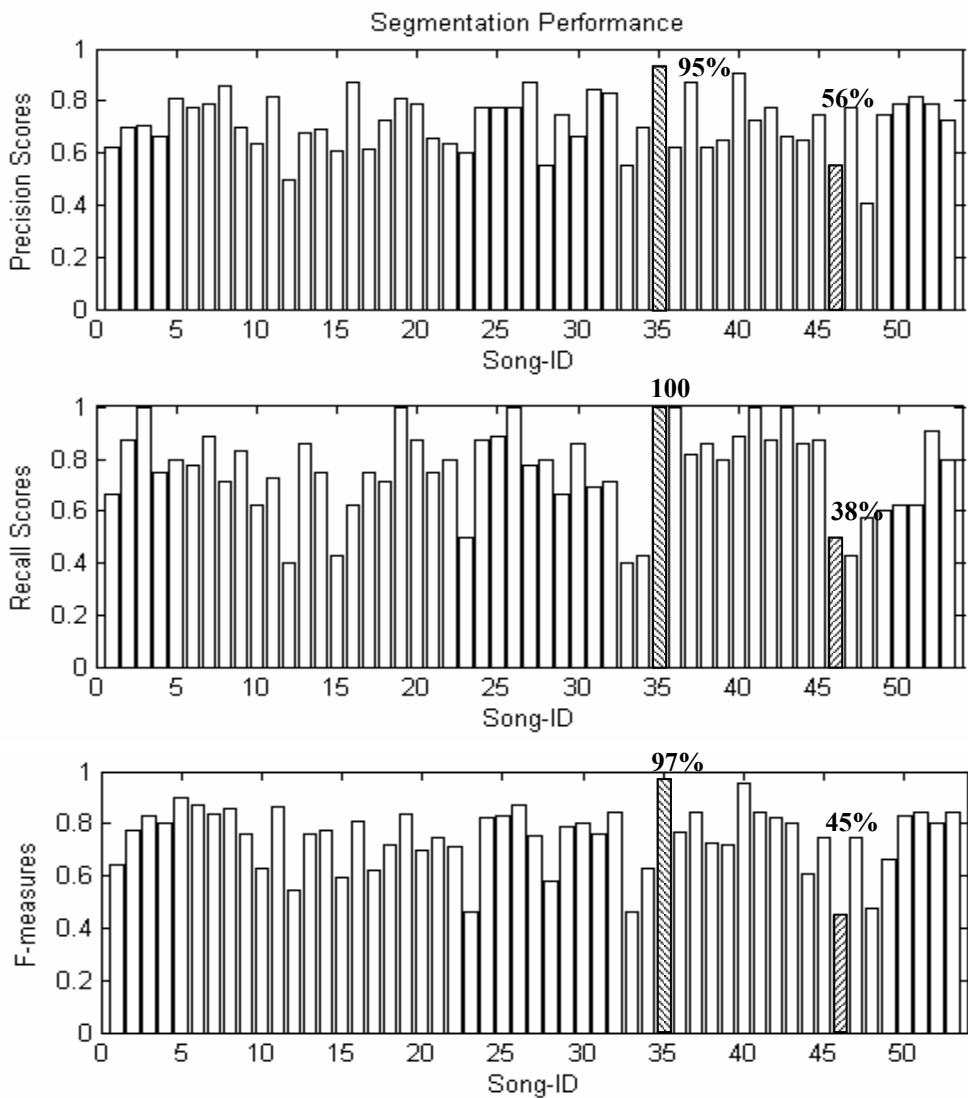


Figure 3.15. The precision, recall and F-measure scores obtained for all songs in The Beatles' test set with a tolerance deviation ± 3 seconds.

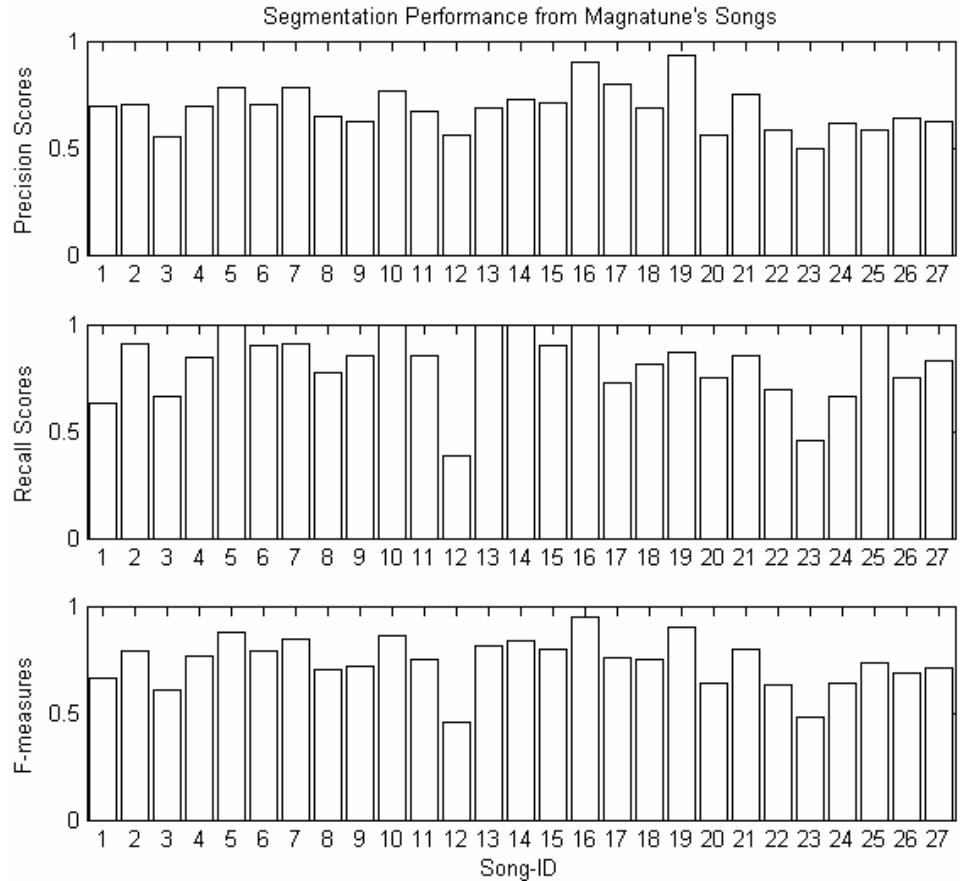


Figure 3.16. The precision, recall and F-measure scores obtained for all songs in Magnatune' test set with a tolerance deviation ± 3 seconds.

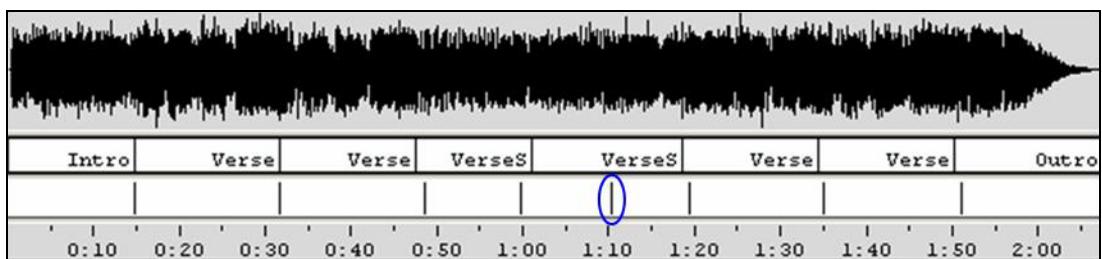


Figure 3.17. Manually labelled segment boundaries (top) and segment boundaries detected by our proposed algorithm (middle) with time position information (below) for SongID-35 entitled *Words of Love*. The label VerseS means an instrumental solo playing the verse. Labels are not yet assigned by the algorithm. Circled segments mark the outliers of the correctly detected segments.

For the second test set, which contains songs from the Magnatune database, our proposed structural changes detection approach achieved an average precision higher than 68% and an average recall of 81% using the ground truth set. The overall F-measure reached 74%, which is quite identical with the one obtained using The Beatles' database.

Significance of the Proposed Audio Descriptors

To investigate the significance of our proposed audio features that are not commonly used in the segmentation task, we evaluate the segmentation results obtained using various combinations of audio descriptors. Table 3.2 shows the use of different combinations of audio descriptors according to our proposal in Table 3.1, together with their labels appearing in the plot given in Figure 3.18.

Label	Descriptors in Phase 1	Descriptors in Phase 2
A	All	All
B	Only MFCC	All
C	All	All, except high-medium energy
D	All	All, except low-bass energy

Table 3.2. Various combinations of the audio descriptors together with their labels appearing in Figure 3.18.

Figure 3.18 illustrates the segmentation performance according to various combinations of audio descriptors. As shown in the graph, with an additional of sub-band energy in phase 1 instead of MFCC features, it shows a slight improvement of 1.46% to the overall effectiveness of the segmentation task. While comparing to low-bass energy features (D), the absence of high-medium energy in phase 2 (C) shows a much greater degradation to the performance of our proposed segmentation algorithm. The impaired overall F-measure reaches as high as 8.1%, with the deficient average precision and recall rates of 8.2% and 8.0%, compared to 2.86% for the low-bass energy features. This may be due to the reason that high-medium energy, which captures spectral content within the frequency ranges of 1.6 KHz to 4 KHz, has also subsumed singers' formant [Sundberg77] properties within it. Thus, it is useful in identifying significant changes in the singing voices for sung music.

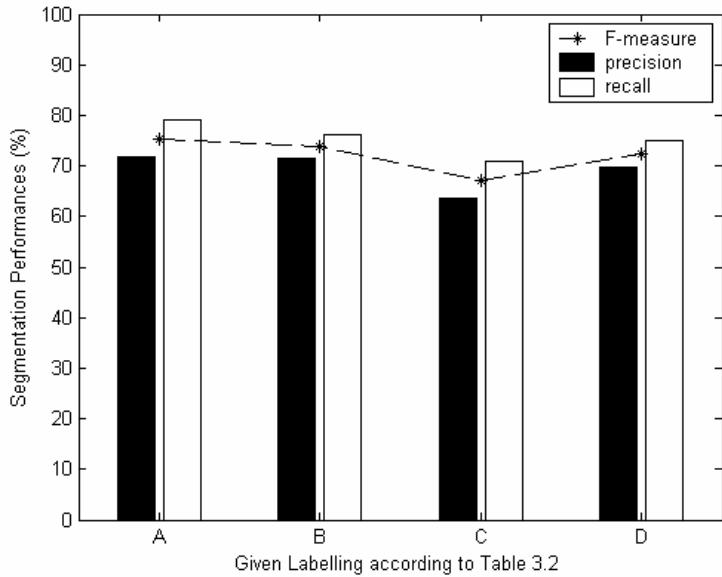


Figure 3.18. The segmentation performance using various combinations of audio descriptors.

Beat Detection Application

Theoretically, by applying reliable beat detection processing to the segmentation algorithm, it should somehow improve the overall detection task. This is because most of the structural changes in popular music appear on the beat. In addition, it can be easier or more convenient to listen to segments that start on the beat (upbeat or downbeat). Thus, to investigate the applicability of beat information to segment detection, we incorporate a beat detection algorithm [Gouyon03] into our system, according to the overview block diagram shown in Figure 3.19. We group the computed frame-by-frame feature vectors according to the induced beat information, instead of using a constant number of frames grouping. Figure 3.20 illustrated the segmentation performance with and without the application of beat detection using The Beatles dataset

From the obtained results, we observe modest improvements at the lower tolerance deviations, whereas no further advancement appears at the extended tolerance deviations. This is due to the inter-beat intervals detected in all songs in the test set. Figure 3.21 illustrates the histogram of 20 equally spaced average inter-beat intervals (IBI) detected from The Beatles' test set. The detected inter-beat intervals from The Beatles' test set have a mean value of 0.6 seconds with a standard deviation of 0.2 seconds with 75% (or third quartile) of the average inter-beat intervals falling below 0.75 seconds. Thus, by applying beat detection processing to our segmentation algorithm, it improves the overall segmentation performances at the tolerance deviations that are approximately within the range limits of the detected inter-beat interval of all songs in the test set. In the case of using The Beatles' as our test set, the tolerance deviations of the improved segmentation results are less than 0.9 seconds, which is consistent with the inter-beat interval descriptions of the test set. This observation leads us to the

assumption that the overall performance might improve at the larger tolerance deviations if reliable detection processing, which identifies the structural units on a larger time scale beyond the inter-beat interval level, is applied to the segmentation algorithm. In this context, efficient music phrase detection or bar detection processing would be very useful in further improving the segmentation task.

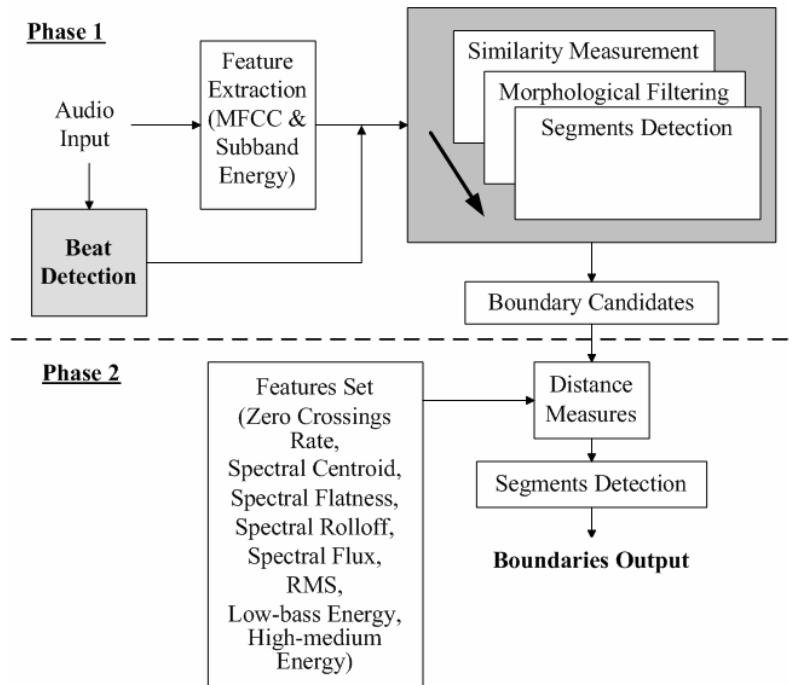


Figure 3.19. Overview block diagram of our approach with the application of beat detection algorithm.

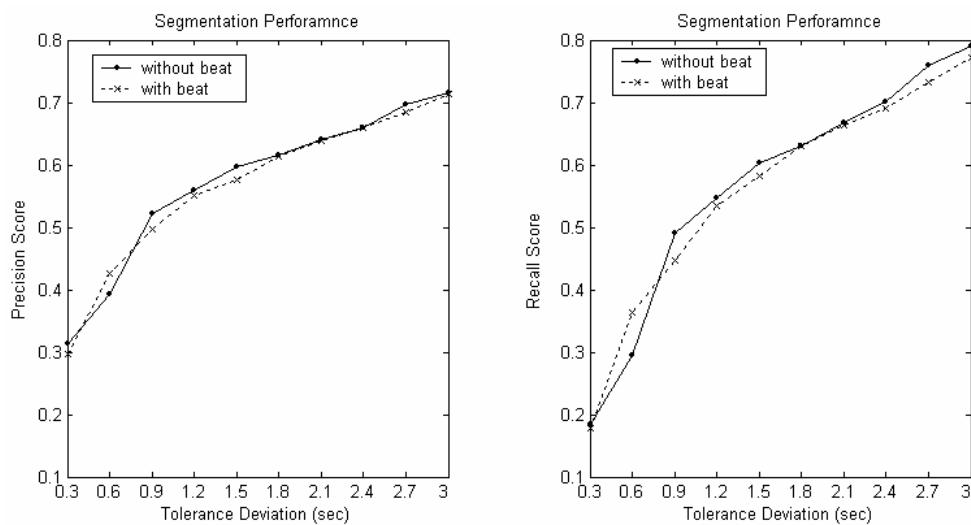


Figure 3.20. The segmentation performance with and without the application of beat detection from The Beatles' test set.

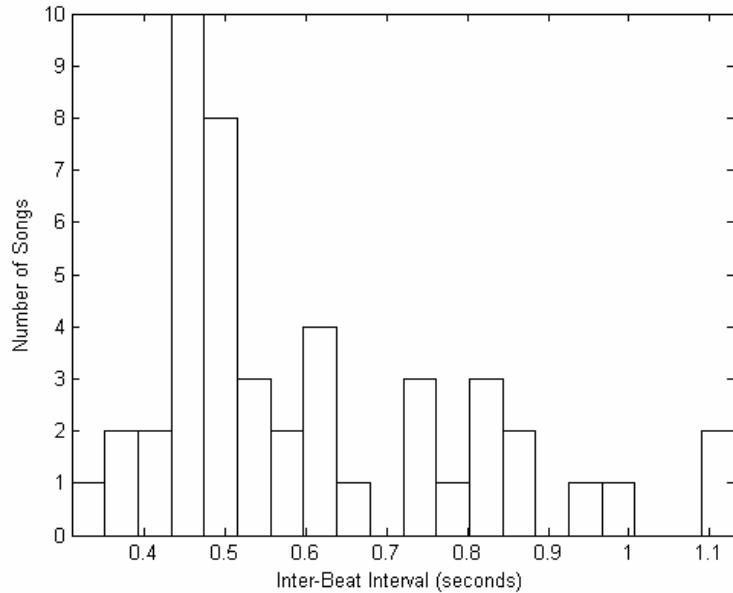


Figure 3.21. The histogram of the average inter-beat interval detected in all songs in The Beatles' database.

From observing the performance of our semantic segmentation algorithm on both test databases, we have noted the fact that precision and recall rates are particularly low for those songs that comprise of smooth transitions between sections. It seems that our descriptors are not sensitive enough to mark these changes. On the other hand, songs with abrupt transitions between sections usually achieve a better rate. Thus, the use of some other descriptors may perhaps contribute towards coping with this issue.

3.3. Summary

In this chapter, we have presented a novel approach for detecting structural changes in music audio using a two-phase procedure with different descriptors for each phase. A combination set of audio descriptors has also been shown useful in detecting music structural changes. In addition, we have explored some other possible ways, such as coupling beat detection into our proposed system, to enhance the segmentation task. Evaluation results have shown that our approach is both valid and improves performance of the segmentation task.

Inter-relationships (e.g. similarities or differences) between the structural segments would give a better grasp of the music structural information directly from the acoustic signals. Comprehending such information would definitely facilitate in efficient handling of huge amounts of music data. A

system with a good semantic segmentation is highly useful for allocating structural changes in music. However the lack of enclosed inter-relationship information with regards to the music segments in audio semantic segmentation may not be useful in practice when dealing with search or retrieval of huge numbers of music audio files. In the next chapter, we present our approach in identifying and extracting music structural descriptions with labelling and time stamping marking (dis) similar sections appeared in the music signals. The system simultaneously identifies repetitive patterns that appear in music and provides a visual representation of the music structure. The following chapter includes an objective evaluation of the performance of our proposed identification method with the use of a mixture of polyphonic music recordings.

Chapter 4

Music Structural Analysis Based on Tonal Features

In Chapter 3, we described a method for segmenting music audio signals. However, a system that can detect structural boundaries in music may not be sufficient for practical use when dealing with search or retrieval of very large numbers of music audio files. In fact, the structural information subsumed in the music signal is beneficial for further applications. Humans assimilate information at a semantic level with remarkable ease. Studies of memory support the assertion that people make use of special landmark or anchor events for guiding recall [Shum94] [Smith79] [Smith78] and for remembering relationships among events [Davis88] [Huttenlocher97]. In our study, we assume that such an assertion also applies to humans in the case of remembering music - we do not recall what we hear in its entirety but through a small number of distinctive excerpts (e.g. chorus, verse, intro, etc.) that have left an impression on our minds. It is usually the case that we only need to listen to one of those distinctive excerpts in order to recall the title for the musical piece, or, at least, to tell if we have heard the song before. Thus, we hypothesize that identifying and extracting music structural descriptions from audio signals would be a primary step towards generating higher-level music metadata, contributing to better and more efficient retrieval of massive amounts of digital audio data. In addition, it will serve as indispensable processing towards music summarization applications that aim to generate abstracts from music audio similar to trailers or previews of movies. In this chapter, we present our approach towards structural analysis of music signals based on tonal features.

We have implemented our own system to perform the structural discovery task via inferring the repeated patterns that appear in music. Our structural description system presented in this chapter is based on Goto's method [Goto03a] for detecting chorus sections in music. We have introduced further improvements on this method to offer a more complete music structural description. This is done by

marking (dis)similar sections that appear in the music signal (i.e. verse, chorus, bridge, etc.) through labelling and time-stamping. Our system takes a polyphonic music audio signal as input and detects repetitions through comparing the pitch chroma information of the music according to frame-to-frame information. After some processing of these detected repetitions, which will be described in detail in the following sections of this chapter, the system will generate transcription files which comprise the beginning and ending times of each sections appearance in the music, together with their given labels. The system is designed to achieve two goals:

- (i) To identify and extract music structural information from audio signals;
- (ii) To visualize all the repetitions that appear in a piece of music. This is to give a visual presentation of music and show key frames of important scenes that occur in the music.

Current literature uses both timbre-related and tonal-related features in discovering the structural descriptions of music. We acknowledge the significance of timbre-related features (i.e. MFCC, etc.) in revealing significant structural changes in music signals. However, our approach in discovering structural descriptions requires independence with respect to timbre and instruments played to reveal repeated patterns in music. In other words, our approach should be able to produce high similarity scores when comparing similar repeated segments but played by different instruments. For this reason, we only consider tonal-related features for performing structural analysis of music signals.

This chapter introduces our proposed system in detail and presents an evaluation of the system's performance for each approach used in the identification task. In Section 4.1, we present an overview of the system. In its subsections, we present in detail the descriptions of all processes carried out by the system. Section 4.2 presents a set of experiments performed on the system and their evaluation results.

4.1. Approach

Figure 4.1 below illustrates the overview framework of our music structural description system. As shown in Figure 4.1, our system involves 9 main processes, each undertaking a different task. These main processes are as follows:

- (1) Feature extraction: segment input signal into overlapping frames of fixed length and compute a set of audio features to describe audio content for each frame;
- (2) Similarity measurement: compute similarity distance between each frame using selected audio features to measure the (dis)similarity between one frame and its neighbouring frames;

- (3) Pre-processing: remove redundancies and enhance information supplied by the similarity representation for later processing;
- (4) Repetitions detection: identify all repeated line segments that appear in music according to the similarity representation;
- (5) Line segments integration: organise all the repeated line segments and recover undetected repeated segments in previous detection processes;
- (6) Repetitive segments compilation: assemble the repeated line segments to construct a comprehensive structural description of music;
- (7) Boundaries adjustment: improve the boundary accuracy of the structural description based on semantic audio segmentation;
- (8) Modulation detection: detect modulation within the song by comparing line segment with ring shifting feature vectors;
- (9) Structural Description Inference: infer music structural description based on time constraint.

The following subsections explain each process in detail.

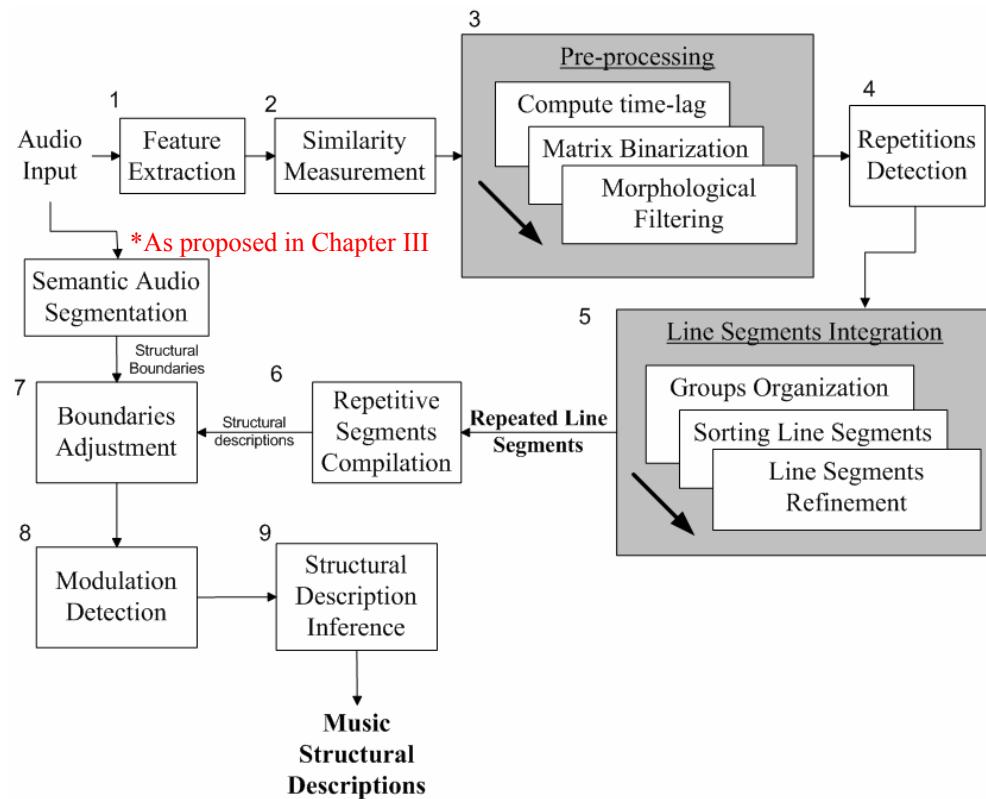


Figure 4.1. Overview framework of our music structural description system.

4.1.1. Features Extraction

Discovering music structure is a key issue in structural analysis research. Hence, extracting a kind of music representation from the audio signal is crucial in discovering the structure of music. Extracting symbolic score-like representation in music could be a possible way to complete the task. However due to the high constraint of present sound source-separation technologies, extracting symbolic representations of polyphonic music from raw audio signals is practically infeasible at the present time. Otherwise, extracting low-level representations of audio signals for musical content description is found to be an alternative way for completing this task.

As mentioned earlier, melody has played an important role in music perception and music understanding with the implicit information that it carries. In fact, perceptual research studies [Dowling78, Edworthy85, Croonen94] have confirmed that contour can serve as a salient cue to melody recognition. Thus, in our approach towards identification and extraction of music structural descriptions, we exploit melody-related features to first find the repeated patterns appearing in a music signal. Our audio input signals consist of polyphonic popular music with the presence of simultaneous pitches from instruments plus voices. Therefore, melody cannot be reliably extracted from polyphonic audio without much error. For this reason, instead of extracting detailed melodic information from the music signals, we revert to using pitch-chroma as a rough approximation of it. We hypothesize that extracting melody-related features focused on the pitch-chroma dimension (i.e. Harmonic Pitch Class Profile, etc.) would be an appropriate manner to deal with our input signals and identify significant musical content.

In the current literature, there exist a few comparisons between timbre-related and tonal-related features for music structural analysis. So far, most of these comparisons [Bartsch05] [Lu04] [Bartsch01] have shown that tonal-related features are better than timbre-related features in discovering the structure of music. However, there exists no specific study in identifying the applicability of different tonal-related features in music structural analysis. In our study, we compare the applicability of tonal-related features generated using two different methods, the Discrete Fourier Transform and the Constant-Q Transform, to reveal repeated patterns in music for music structural analysis. Table 4.1 below shows the grouping of the compared tonal-related features.

Discrete Fourier Transform	Constant Q Transform
Harmonic Pitch Class Profile (HPCP) Pitch Class Profile (PCP)	Constant-Q Profile (Cq Profile) Constant-Q (CQP)

Table 4.1. The list of tonal-related descriptors generated using two different methods for structural description discovery.

Our tonal-related features are computed by mapping frequencies to pitch class values for a single octave. All the compared tonal-related features have an interval resolution of one third of a semitone (chroma), with the size of the pitch class distribution vectors equal to 36. As mentioned in the beginning of this chapter, our approach in discovering structural descriptions requires features that are highly sensitive to tonal similarities and independence with respect to timbre and instruments played to reveal repeated patterns in music. Thus, different from Lu et al.'s proposed features in music structural analysis [Lu04], we use octave mapping for all our compared features, including both tonal descriptors computed using the constant Q transform. This is due to the reason that through octave mapping, the CQT features are more sensitive to tonal similarities compared to the non-octave mapping of the features. Figure 4.2 illustrates self-similarity matrices of three notes based on cosine distances among three notes, which includes B4 played by the bassoon (*B_B4*), B4 by the clarinet (*Cl_B4*), and C5 by the bassoon. The similarity plots are normalized to [0,1], and the brighter points represent high similarity. From the similarity plots, it is noted that the similarity score between B4 played by the bassoon (*B_B4*) and B4 played by the clarinet (*Cl_B4*) is higher for the octave-mapped constant Q transformed features than the non-octave-mapped features. Thus, the act of octave mapping on our used tonal descriptors is considerably fulfills the features properties required by our approach. Another advantage of using octave-mapped tonal features is that tonal modulations, which occur within a piece of music, could be easily detected by ring shifting the tonal features (as to be explained in Section 4.1.8). Thus, for analyzing the structure of music, we adopt the octave mapping procedure for all our features.

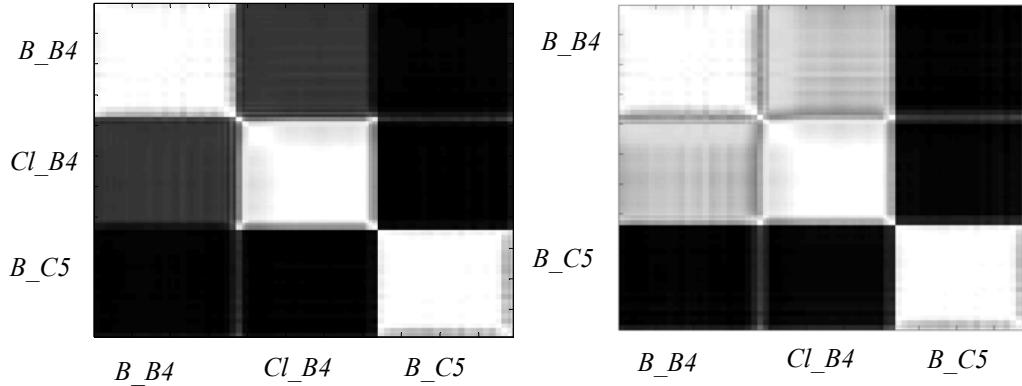


Figure 4.2. Self-similarity matrices of three notes, which include B4 played by the bassoon (*B_B4*), B4 by the clarinet (*Cl_B4*), and C5 by the bassoon (*B_C5*), using different Constant-Q feature vectors: (right) Constant-Q extracted directly from 5 octaves of musical notes (left) Constant-Q extracted from 5 octaves of musical notes and mapped into 1 octave.

As in most content-based analysis, our system first requires the short-term descriptions of the input audio signal. The input signal is a complete full-length music audio signal. We segment the input signal into overlapping frames (4096-samples window length) with the hop size of 512 samples. This is then followed by extracting pitch class distribution features for each of these frames. Here, we use one of three different approaches for extracting low-level tonal features from input signals. The general block diagram for computing pitch class distribution features is shown in Figure 4.3.

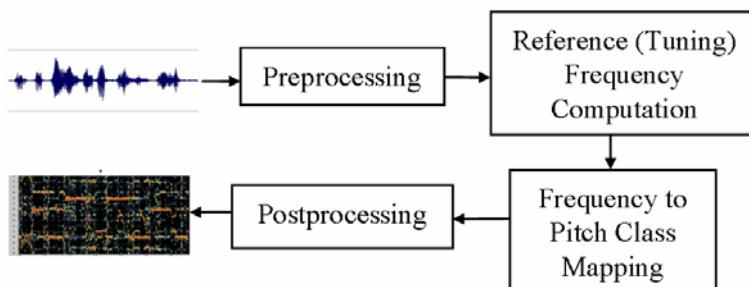


Figure 4.3. General diagram for computing pitch class distribution features.

We focus here in describing the main differences between the four different approaches: Constant Q Spectral Transform Profiles (CQP), based on [Brown91], Constant Q Profiles (CQ-Profiles), as proposed in [Purwins00], Pitch Class Profiles (PCP), as proposed in [Fujishima99] and the Harmonic Pitch Class Profiles (HPCP), explained in [Gómez06a].

Pre-processing

CQP and CQ-profiles use the constant-Q transform as a preprocessing step before mapping frequencies to pitch class values, while PCP and HPCP use the DFT. For both PCP and HPCP, the preprocessing step also includes a frequency filter after DFT, so that only the frequency band between 100 and 5000 Hz is used. HPCP finally includes a peak detection procedure, so that only the local maxima of the spectrum are considered. More details regarding the HPCP computation can be found in [Gómez06b].

Reference Frequency Computation

A reference frequency computation procedure is used before computing HPCP, in order to estimate the deviation with respect to 440 Hz of the frequency used to tune the piece. This is done by analyzing the deviation of the peak frequencies with respect to the perfect tuning. PCP, CQ-profiles and CQP use a fixed frequency grid with a 440 Hz reference.

Frequency to pitch class mapping

Once the reference frequency is known and the signal is converted into a spectrogram by means of DFT or constant-Q analysis, there is a procedure for determining the pitch class values from frequency values. In the case of CQP and CQ-profiles, the weight of each frequency to its corresponding pitch class is given by the spectral amplitude, whereas the PCP and HPCP use the squared value. The HPCP introduces an additional weighting scheme using a cosine function (described in [Gómez06a]), and considers the presence of harmonic frequencies, taking into account a total of 8 harmonics for each frequency. In the four compared approaches, the interval resolution is set to one-third of a semitone, so that the size of the pitch class distribution vectors is equal to 36.

Post-processing

Finally, in the case of HPCP and PCP, the computed features are normalized frame-by-frame dividing through the maximum value to eliminate dependency on global loudness. Table 4.2 shows the computation methods of the compared tonal features.

4.1.2. Similarity Measurement

The second process in our system is similarity measurement between each frame feature vectors. First, the system uses the computed HCPC feature vectors as input and selects a set of candidates for later processing. The candidate set consists of the first frame feature vectors from every 10 frames, with each representing the pitch class distributions of approximately every 116 millisecond of the original input signal. Here, we only consider the first frame feature vectors instead of all 10 frames. There are two reasons for us to do so. First, it prevents our system from having high computational cost by

processing the complete HPCP features of the input audio signal. Second, we assume that there are not much significant changes in terms of music context within such a short interval. Thus, taking the first frame features of every 116 milliseconds interval is sufficient to identify significant changes within the music. From these candidates, we measure the (dis)similarity distance between each candidate, $v(n)$, to its neighbouring candidates using the cosine similarity function. The cosine similarity function calculates the dot product of the features vectors, normalized by their magnitudes. It produces distance measures within the range of 0 to 1. The similarity score tends towards 1 when there is a strong similarity between two candidates. In reverse, the similarity scores tend towards 0 when there is less similarity between two candidates. The cosine of the angle is given by the expression:

$$SD(X, Y) = \frac{X \cdot Y}{\|X\| \|Y\|} \quad (4.2)$$

We then embed the computed (dis)similarity distance values in a two-dimensional representation plot to reveal the repeated patterns that occur in the musical structure of the input signal. Figure 4.4 illustrates a two-dimensional similarity plot of the song entitled *I'm a Loser* with the use of HPCP descriptors. Repeated patterns in music are visible as bright off-diagonal lines running from top left to bottom right.

Method	Pre-processing	Reference frequency computation	Frequency to Pitch Class Mapping	Post-processing
HPCP [Gómez06a]	DFT (100-5000Hz), transient detection	Analysis of peak deviation	Square of spectral magnitude and weighting scheme	Normalization
PCP [Fujishima99]	DFT (100-5000Hz)	No	Square of spectral magnitude	Normalization
Cq profiles [Purwins00]	Constant Q transform (131.25 Hz - 5 octaves above or ~4000Hz)	No	Spectral magnitude	No
Constant Q transform [Brown91]	Constant Q transform (92 Hz - 5 octave above or ~2900Hz)	No	Spectral magnitude	No

Table 4.2. The different computation methods of the compared tonal-related features.

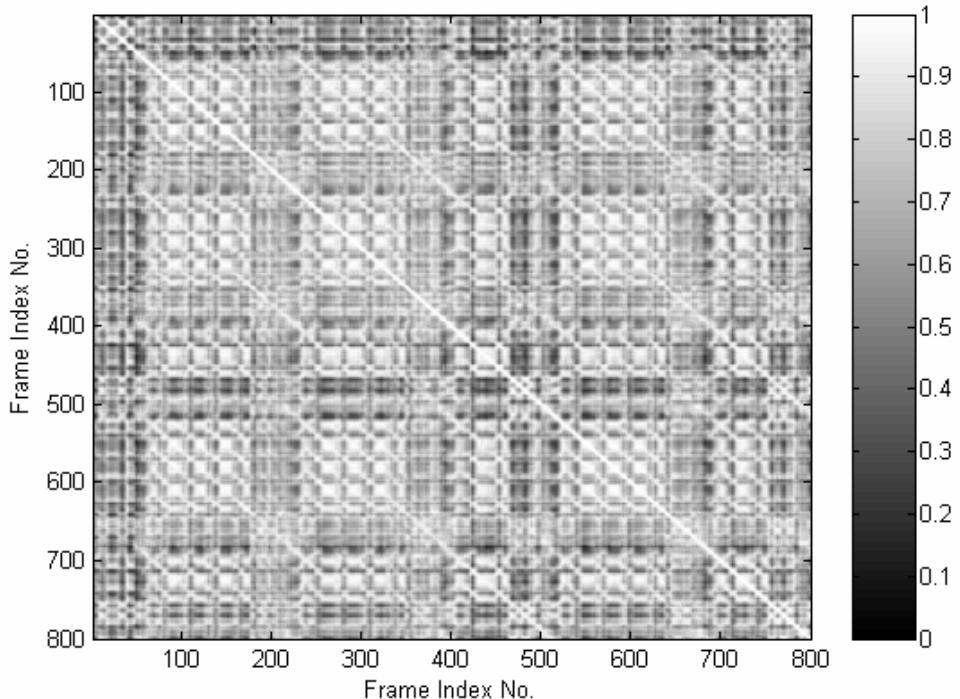


Figure 4.4. Two-dimensional similarity plot of The Beatles' song entitled *I'm a Loser*.

Besides cosine distance, Euclidean distance could be an alternative way for computing similarity. In this chapter, we choose cosine distance over Euclidean distance. The reason lies in the type of audio features we used to discover repeated patterns in music. As mentioned in the beginning of this chapter, our approach in music structural analysis is based on identifying and inferring repeated patterns that appear in music with the use of tonal features. Thus, the success of the pattern recognition task is highly influenced by the high sensitivity of the distance measure with regards to the tonal descriptions subsumed in the computed feature vectors. Figure 4.5 illustrates self-similarity matrices computed based on PCP features of three notes, which includes B4 played by the bassoon (*B_B4*), B4 by the clarinet (*Cl_B4*), and C5 by the bassoon. The similarity plots are normalized to [0, 1], and the brighter points represent high similarity. From the similarity plots, it is noted that the similarity scores between B4 played by the bassoon (*B_B4*) and B4 played by the clarinet (*Cl_B4*) is much higher for the matrix computed using the cosine distance than the Euclidean distance. In this case, it demonstrates that cosine distance is more sensitive than Euclidean distance for our used tonal features. Thus, by using the Euclidean distance to measure the similarity for tonal feature vectors, it may introduce some noise to the later repeated patterns detection processes. Even though our given examples comprise of only single notes this observation should not be much different when applied to chords played by different instruments, especially with our considered tonal-descriptors which are insensitive to timbre differences. Besides, even though similar chord progressions may comprise notes played by different instruments, their tonal descriptions still subsume more or less similar properties.

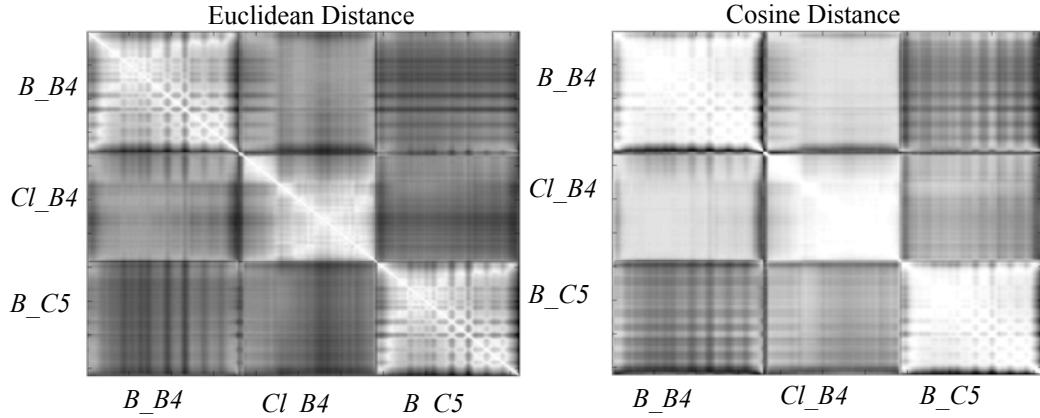


Figure 4.5. Self-similarity matrices of three notes, which include B4 played by the bassoon (*B_B4*), B4 by the clarinet (*Cl_B4*), and C5 by the bassoon (*B_C5*), using difference distance measures: (right) Cosine distance (left) Euclidean distance.

4.1.3. Pre-processing

In order to ease the process of identifying repetitive segments in music, we compute the time-lag matrix of the similarity representation, computed from the previous processing, by orientating the diagonal of the computed similarity matrix towards the vertical axis. The rotated time lag matrix, $L(l, t)$ between chroma vector $v(t)$ at time t and at time lag l , $v(t-l)$, is defined as

$$L(l, t) = SD(v_t, v_{t-l}) \quad (4.3)$$

Figure 4.6 illustrates the converted time-lag matrix with the x-axis referring to the lag and the y-axis referring to the time. The vertical lines, which appear to be parallel to the y-axis in the time-lag matrix plot indicate the repeated segments that appear in the music. For instance, a vertical line from $L(15, t_{begin})$ to $L(15, t_{end})$ in the time-lag matrix denotes that the audio section between t_{begin} and t_{end} seconds is a repetition of the earlier section from time $(t_{begin}-15)$ sec to $(t_{end}-15)$ sec. The length of each line segment appears in the time-lag matrix plot indicating the duration of each repeated segment in the music. In other words, line segments with long vertical lines signify long repetitions of music segments and vice versa. Hence by detecting the vertical lines that appear in the time-lag matrix, we can obtain all the repetitions that appear in the music signal.

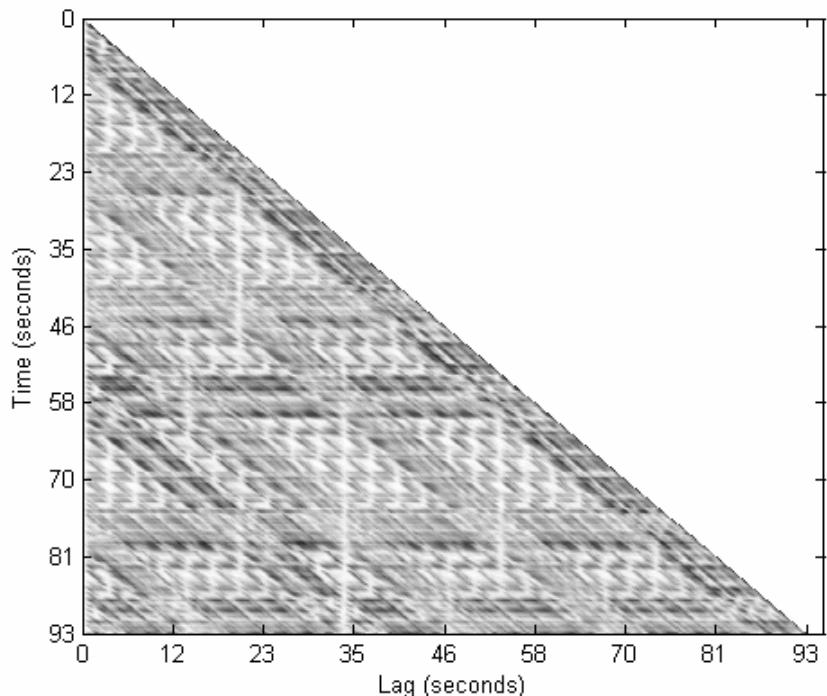


Figure 4.6 illustrates the time-lag matrix, L , for the song *I'm a Loser* by The Beatles with its x-axis referring to the lag and its y-axis referring to the time.

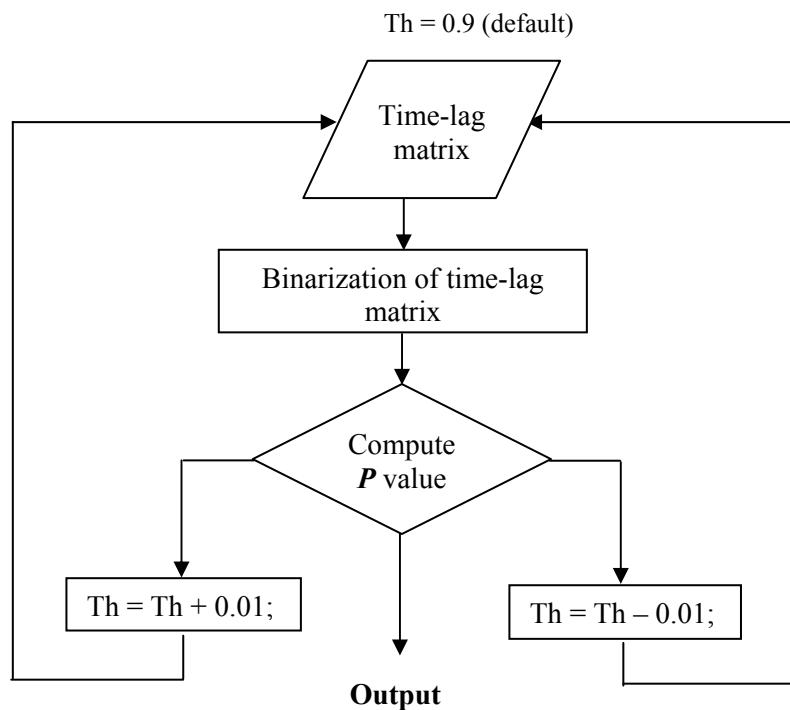


Figure 4.7. Flow chart illustrating the iterative binarization process.

As shown in Figure 4.6, there exists much noise in the time-lag matrix. Hence, in order to detect repetitions or vertical lines in the time-lag matrix, we would like to consider only a certain degree of similarities. Using a fix degree of similarities is not practical when dealing with broad categories of audio input signals, which may have differences in recording quality and etc. Thus, in our approach, we perform a binarization process on the time-lag matrix based on an adaptive threshold to remove redundancies. The threshold, Th , used for the binarization process will decide the degree of similarity measures to be retained from the matrix for later processing. The implementation of the binarization process is based on an iterative procedure as shown in Figure 4.7.

For initialization, our adaptive threshold holds a default value of 0.9. It means that we would only consider similarity measures with values more or equal to 0.9. We first binarize the similarity measures in the time-lag matrix using the default threshold value. In another words, those similarity measures which are less than the threshold value are set to 0 whereas those higher or equal are set to 1. Then we compute a P value from the binarized matrix to evaluate the sufficiency of information it retained. The P value is defined as:

$$P = \frac{\text{total number of 1 in time-lag matrix}}{0.5 \times \text{Area}(\text{time-lag matrix})} \quad (4.4)$$

Based on the computed P value, we consider three cases as listed below:

- (i) If $P > 0.039$, increase the threshold by 0.01 and return to the beginning of the procedure;
- (ii) Else if $P < 0.02$, reduce the threshold by 0.01 and return to the beginning of the procedure;
- (iii) Else, quit the iterative process and output the binarized time-lag matrix.

For the first two cases, the iterative procedures continue with alteration to the threshold value. The whole iterative procedure only terminates when the third case is fulfilled. In such a case, the system will output the binarized time-lag matrix and move on to the last operation in the pre-processing section. The values used in evaluating different cases in P are empirical results obtained through observing various input signals. When $P > 0.039$, it denotes superfluous information in the binarized time-lag matrix. Thus, by increasing the level of threshold, P , it removes the redundancies. When $P < 0.02$, it denotes that insufficient information is contained in the binarized time-lag matrix. Thus, it is necessary to reduce the level of threshold, P , in order to yield more information for further processing. Figure 4.8 illustrates an example of binarized time-lag matrix.

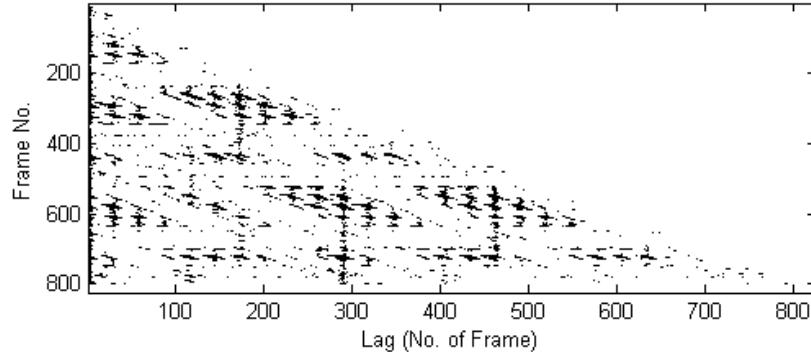


Figure 4.8. An example of binarized time-lag matrix.

The last operation within the pre-processing section consists of applying the opening operation of the morphological filter, a widely used filter operation in image processing, to the binarized time-lag matrix. Please refer to Chapter 3 for details regarding the operations of the morphological filter. The functionalities of the opening operation in our application are:

- (i) to separate vertical line segments, which contain large in-between gaps, to several short segments;
- (ii) to remove line segments, which are too short to contain any significant repetitions of music.

As shown in Figure 3.5.b presented in Chapter 3, we can clearly see how the ‘Opening’ operation opens the gaps within the signal while removing successive ones that are shorter than the structuring element in the one-dimensional binary signal. As mentioned in Chapter 3, Section 3.1.2, the opening operation of the morphological filter is based on erosion and dilation operations [Filonov05]. In general, dilation causes objects to dilate or grow in size while erosion causes objects to shrink. The ‘Opening’ operation works by eroding the signal then dilating the results. The amount of changes (growth or shrinkage) depends on the choice of the structuring element. The following paragraph explains briefly how dilation and erosion work in detail.

As mentioned in Section 3.1.2, dilation works by moving the structuring element over the input signal where the intersection of the structuring element reflected and translated with the input signal is found [Young02]. Figure 3.3.a shows how dilation adds ones to runs of zeros that are shorter than the structuring element. While dilation works by moving the structuring element over the input signal, erosion of the input signal, A, and the structure element, B, is the set of points x such that B translated by x is contained in A [Young02]. In contrast with the dilation operation, the output is set to zero unless the input is identical with the structuring element. Figure 3.3.b shows how erosion removes runs of ones that are shorter than the structuring element.

Here, we treat our binarized time-lag matrix as a one-dimensional non-binary signal. As mentioned before, the opening operation works by eroding the signal followed by dilating the results (as illustrated in Figure 3.4b). Alternatively, we implement the erosion operation, $Er(i, j)$, by first applying a zero-phase rectangular window, $w(n)$, along the perpendicular of the binarized time-lag matrix, $x(i, j)$, and computing the minimum value within each windowed signal. That is,

$$Er(i, j) = \min \{x(i, j + n)w(n)\}, \quad - (N-1)/2 \leq n \leq (N-1)/2 \quad (4.5)$$

where $x(i, j)$ refers to the lag, i and j refers to the time at the y-axis and x-axis of the binarized time-lag matrix. $w(n)$ is the zero-phase rectangular window function, which is used to define the minimum length, N , of relevant line segments such that line segments shorter than this minimum length are to be removed from the binarized time-lag matrix, and are defined as

$$w(n) = \begin{cases} 1, & |n| \leq \frac{N-1}{2} \\ 0, & \text{otherwise} \end{cases} \quad (4.6)$$

We then perform a dilation operation on the eroded signal, $Er(i, j)$, by applying the previously used rectangular window, $w(n)$, to $Er(i, j)$ then computing the maximum value within each windowed signal. That is,

$$Op(i, j) = \max \{Er(i, j + n)w(n)\}, \quad - (N-1)/2 \leq n \leq (N-1)/2 \quad (4.7)$$

Finally, we yield a binarized time-lag matrix with removed vertical line segments that are less than the size of the window used in the morphological filtering operations. Figure 4.9 shows the same example given in Figure 4.8 before and after applying morphological filtering operations. In our study, we have experimented with using different window lengths in order to find the optimal one for our proposed method. The experimental results are reported later in Section 4.3.

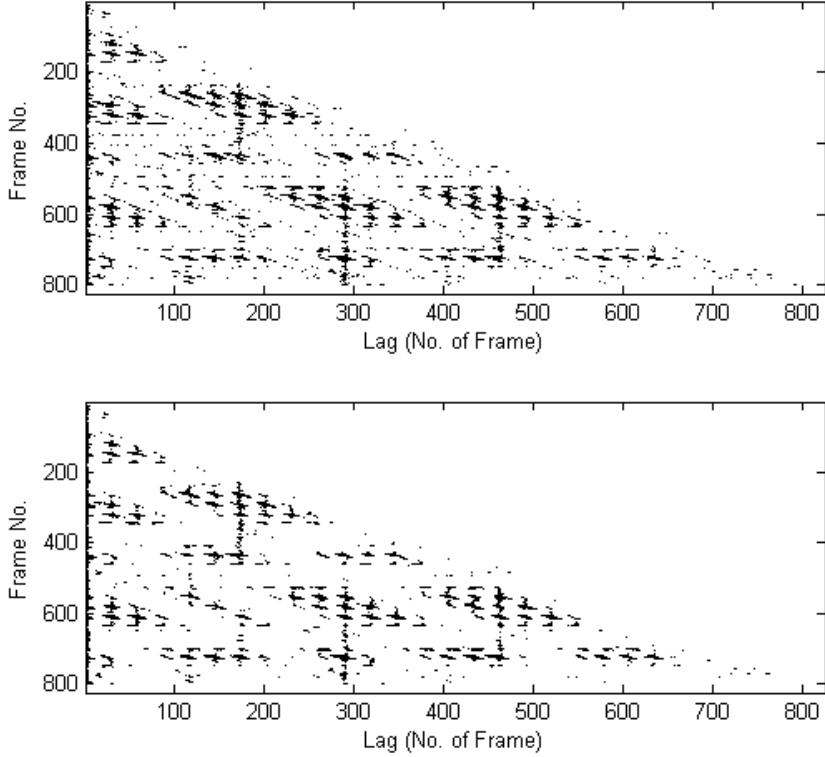


Figure 4.9. Binarized time-lag matrix: before (upper) and after (below) applying morphological filtering operations

4.1.4. Repetition Detection (Listing the repeated sections)

The main goal of this process is to detect repetitive segments. This process requires the output data from the post-processing process, $L_p(l, t)$, as an input signal. As mentioned earlier, vertical line segments in the time-lag matrix represent the occurrence of repetition in music. Thus, for finding the possibility of each lag for containing line segments, $P_r(l, t)$, we sum up each column of the time-lag matrix according to the lag. Since we only consider elements below the diagonal of $L_p(l, t)$ and the number of elements decreases corresponding to the increase of lag, we normalize the summation results with the total number of elements in each lag. The calculation for the possibility of containing line segments, $P_r(l, t)$, of each lag is defined as:

$$P_r(l, t) = \int_l^t \frac{L_p(l, \tau)}{t - l} d\tau \quad (4.8)$$

Figure 4.10 illustrates the possibility of containing line segments, $P_r(l, t)$, corresponding to each lag.

High $P_r(l, t)$ marks frequent repetitions whereas low $P_r(l, t)$ marks that infrequent repetitions occur in lag, l .

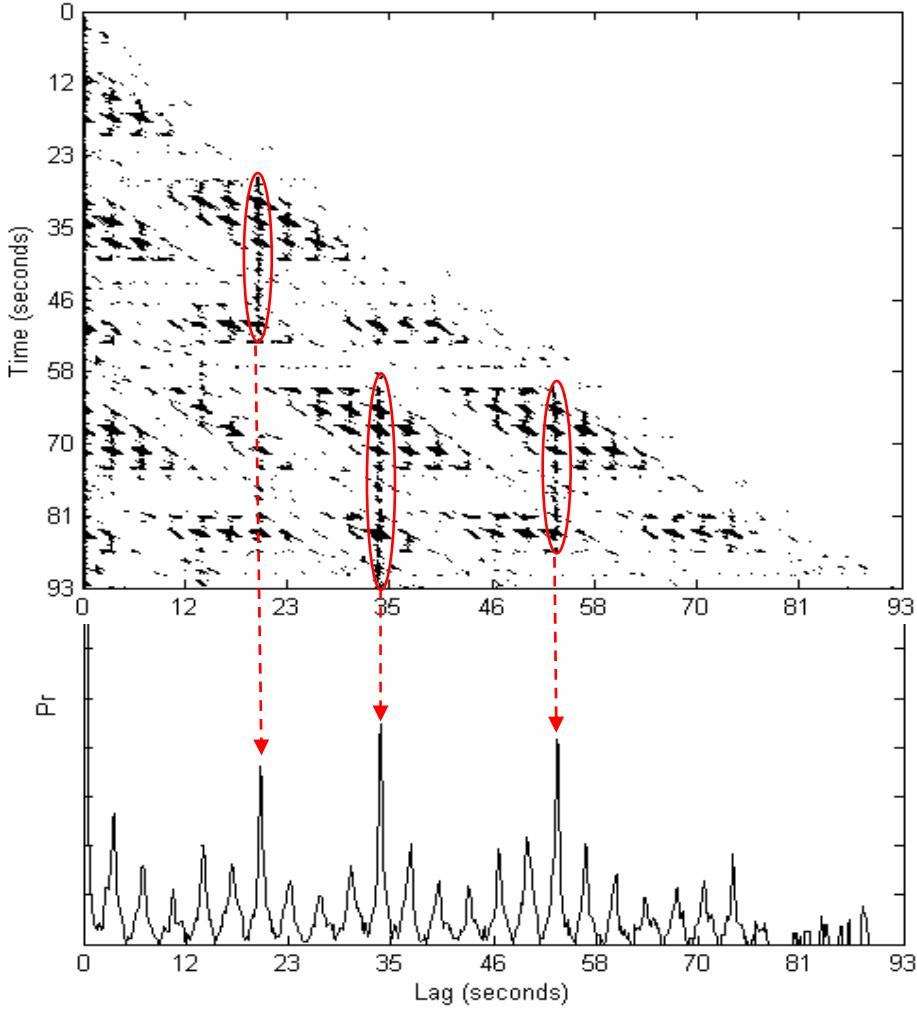


Figure 4.10. The possibility of containing repetition, $P_r(l, t)$, corresponds to each lag.

For searching line segments, we select all peaks appearing in $P_r(l, t)$ and store their lag information in descending order as $l_{PeakSort}$. We then evaluate the occurrence of line segments in $L_p(l, t)$ alternately for each element in $l_{PeakSort}$. We compute $L_p(l_{PeakSort}, t)$ for each $l_{PeakSort}$ and search for the occurrence of vertical line segments. Here, we hypothesize that repetitions, which hold for less than 4 seconds (or less than 2 bars for a music piece with a tempo of 120 beats-per-minute in 4/4 time signature), do not carry much significant musical information. Thus, for detecting repetitive

segments in music, we only consider those line segments with durations longer than 4 seconds. For each detected line segment, we store the beginning and ending time of the repeat segment together with its repeated segment based on $l_{PeakSort}$ information. For instance, when a line segment between $L_p(l_{PeakSort} = l_p, T_1)$ and $L_p(l_{PeakSort} = l_p, T_2)$ is located, it means that a segment between time T_1 and T_2 is the repetition of an earlier segment at time $T_1 - l_p$ until $T_2 - l_p$. Hence, by the end of an iterative detection process, we yield a set of repetition pairs. Pseudo code shown in Figure 4.11 outlines the above mentioned line segments searching algorithm.

```

Select peaks from  $P_r(l, t)$ 
Let  $l_{PeakSort}$  = peaks' lag information in  $P_r(l, t)$ 
Sort  $l_{PeakSort}$  by descending order

FOR each of the  $l_{PeakSort}$ 
    Search line segments appear in  $L_p(l_{PeakSort}, t)$ 

        FOR each of the obtained line segments
            IF length of line segment less than 4 seconds
                Delete line segment
            ELSE
                Store starting time and ending times of line segment
                Store starting time and ending times of repeated line segments
            END
        END
    END

```

Figure 4.11. Pseudo code outlines the line segments search algorithm.

4.1.5. Integrating the Repeated Sections

In this section, we organize the detected repetition pairs obtained from previous steps into groups. Apparently, different line segments that share a common line segment are repetitions of one another, for example repetition pairs A, B and C (according to the y-axis) given in Figure 4.12. Thus, if these segments are to be labelled, they should be given the same labelling.

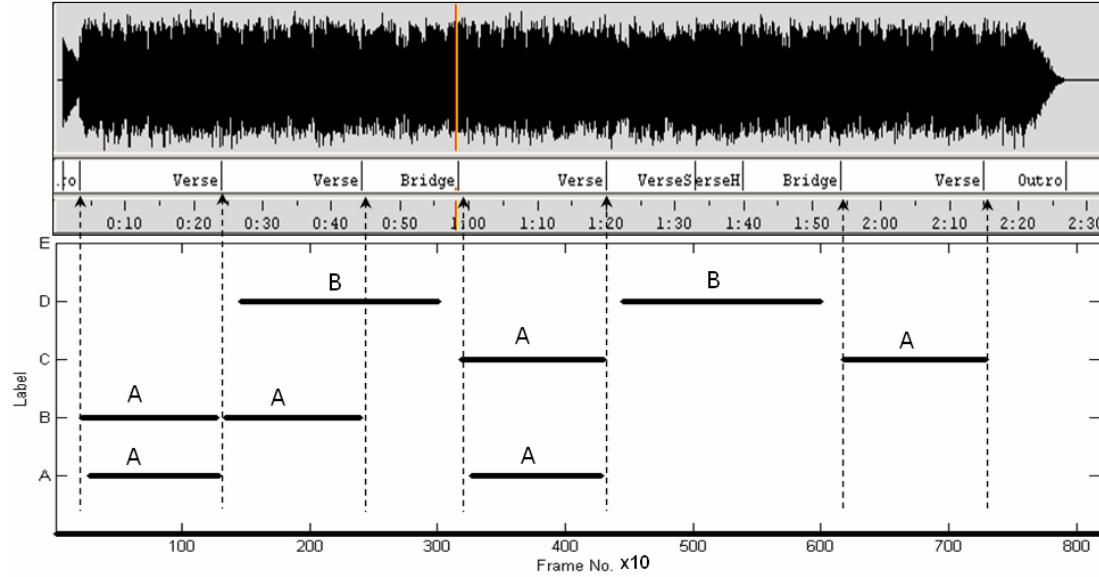


Figure 4.12. Detected repetitions correspond to the ground truth annotation of *A Hard Day's Night*. (Description: y-axis shows the given label of each line segment. Bold labels above each line segment denote the relationship among the line segments).

Based on this observation, we integrate the line segments, which share a common line, into a same labelled group. From this, we yield a set of repetition groups with different labels marking the different repetitive segments appearing in a music piece. That is

$$Group_{repetitions} = \{Group_1, Group_2, \dots, Group_n\} \quad (4.9)$$

where n is the number of repetition groups. In each repetition group, we sort the repeated line segment in ascending order based on their time information, represented as

$$Group_A = \{[Tbegin_1, Tend_1]; [Tbegin_2, Tend_2]; \dots; [Tbegin_m, Tend_m]\} \quad (4.10)$$

where $Tbegin_1 < Tbegin_2 < \dots < Tbegin_m$

$Tbegin$ and $Tend$ denote the beginning time and ending time of the repetitive segments whereas m is the number of repetitive segments in $Group_A$.

For the refinement of line segments, we select the first line segment from each group in $Group_n$, and correlate it alternately with the pre-processed features, $v(n)$, as mentioned in the earlier section 4.1.2. This is for the purpose of recovering undetected repetitions that we have missed in the previous

detection process. We compute the distance measure, $E(n)$, for the selected line segment and a sliding window of the same length through the pre-processed features, $v(n)$. The distance measure, $E(n)$, is defined as

$$E(n) = \sqrt{\frac{\sum \sum (compared_{segment} - v(n)_{len_compared})^2}{len_compared^2}} \quad (4.11)$$

where $Compared_{segment}$ denotes the compared segment features and $len_compared$, its length. The n^{th} pre-processed feature sequence with the length $len_compared$ is represented by $v(n)_{len_compared}$. The computed distance measures are within the range of 0 and 1 with low distance measures indicating strong correlations with the compared segment and vice versa. In fact, when there exists 0 in the computed distance measures, it marks the correlation of the compared segment to itself. Figure 4.13 illustrates the correlation between compared segments with pre-processed features, $v(n)$ corresponding to time. As shown in Figure 4.13, the self-correlated compared segment occurs after 31 seconds of the starting point of the song, marking the actual time position of the compared segment in the input music signal.

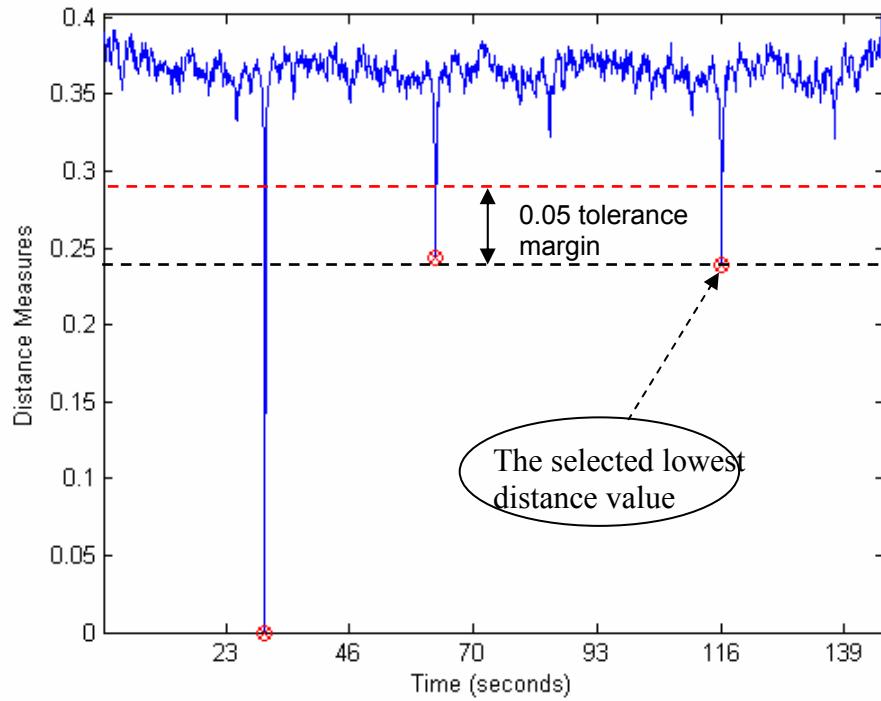


Figure 4.13. The correlation between selected segments with pre-processed HPCP features, $v(n)$. Circled crosses mark the selected local minima based on the computed distances with a predefined threshold.

To detect significant repetitions appearing in music, we use an empirically predefined threshold based on the computed distances. Excluding the distance of the compared segment to itself (which is always zero), we select the lowest occurring distance value. To obtain the predefined threshold, we add a tolerance margin of 0.05 to this value. Then, all local minima falling below the threshold are considered relevant to the occurrence of repetitions. We sort the considered local minima based on the distance measure in descending order. With the length of the compared segment, we estimate and store the corresponding beginning time and ending time for each considered local minimum and form a set of candidate segments. We hypothesize that repetitions of a segment do not overlap with each other. Hence, we disregard those candidate segments that overlap with any of the line segments in the group that hold the same label as the compared segment. The remaining ones are labelled and included in the correct group as omitted repetitions from the earlier detection process. We then reorganize line segments in the group with an ascending order based on their time information. This is similar to the earlier sorting processes of the line segments for each group in $Group_n$. Pseudo code shown in Figure 4.14 giving a rough outline of the above mentioned refinement algorithm in recovering omitted repetitions from the earlier detection process.

```

FOR each repetition group in  $Group_n$ 
    Let  $segments\_inGroup$  = line segments in the repetition group
    Find lowest distance value besides zero

    Let  $lowest\_distance$  = lowest distance value
    Select local minima within  $lowest\_distance + 0.05$ 
    Sort selected local minima in descending order based on distance value

    Let  $Z$  = length of a line segment in  $segments\_inGroup$ 
    FOR each selected local minima
        Compute starting time and ending time of local minimum based on  $Z$ 
        Store starting time and ending time of local minimum
        IF overlapping with any  $segments\_inGroup$ 
            Remove selected local minimum
        ELSE
            Label and insert selected local minimum in  $segments\_inGroup$ 
            Sort  $segments\_inGroup$  in ascending order based on time
        END
    END
END

```

Figure 4.14. Pseudo code outlines the line segment refinement algorithm.

4.1.6. Repetitive Segments Compilation

For generating the music structural description, we select the three most repetitive groups, $Group_n$, (i.e. with the highest number of elements). We compile the repetitive segments by lining up all the line segments of these repetitive groups according to their labels as shown in Figure 4.12. If there exists an

overlap between two particular labels (e.g. A and B as shown in Figure 4.15), all the overlapped sections of these two labels will be given a new label (e.g. C), whereas the non-overlapped sections will be given another label (e.g. D). Unlabelled sections between all the labeled segments (e.g. E and F) will each be given a new label as a new unique repetition group. We then select one line segment of each label and perform another repetition detection procedure by correlating it with the pre-processed features, $v(n)$, as described in Section 4.1.5, this time with the goal of finding all the corresponding repetitions that appear in the music signal. Finally, the repetition detection process terminates when we have checked all labels obtained from the previous operation.

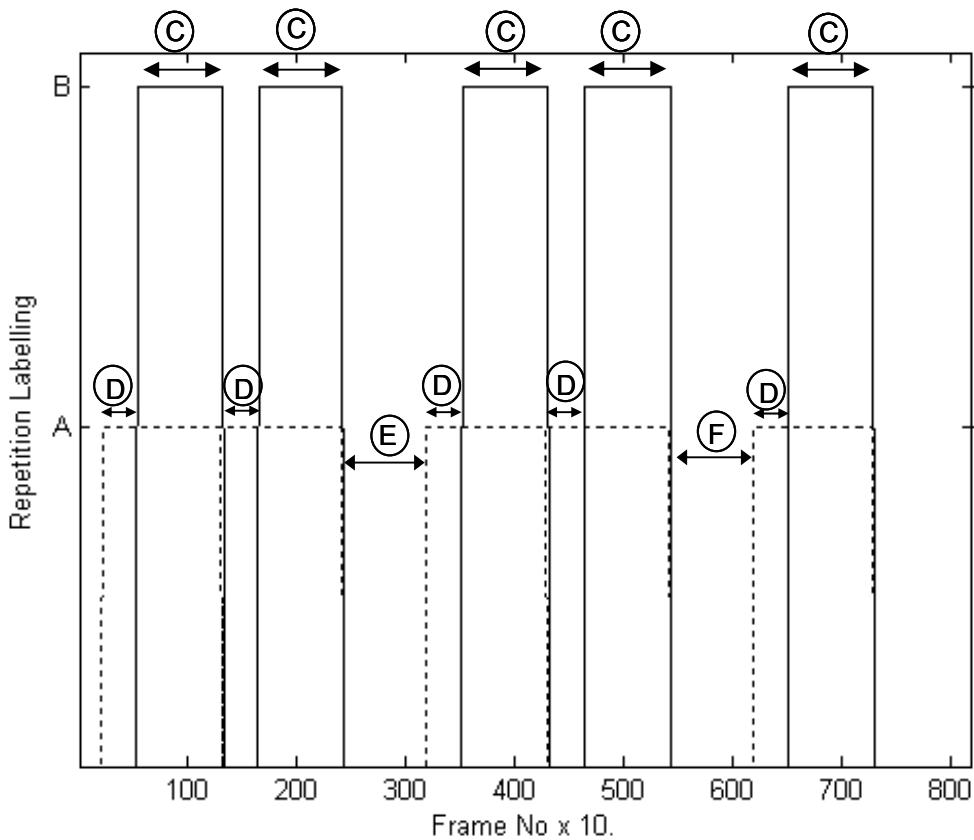


Figure 4.15. Repetitive segments compilation process with generated new labels. (Descriptions: Labeling given in the y-axis denotes the original repetition labeling before the repetitive segments compilation process. Circled labels mark the generated new labels after the repetitive segments compilation process)

4.1.7. Boundaries Adjustment based on Semantic Audio Segmentation

In the previous chapter, we have explained our semantic audio segmentation algorithm for detecting the significant abrupt changes in the audio contents. In our semantic audio segmentation approach, we

hypothesize that abrupt changes in audio content occur where there is a sectional transition (e.g. intro → verse, verse → chorus, etc.) in the music signal. Thus, unlike our current approach, semantic audio segmentation focuses on the significant deviations in the audio content for finding structural changes of music signals. As presented in Chapter 3, semantic audio segmentation has provided a way to segment music signals to a larger extent compared with beat detection and onset detection algorithms. In other words, the detected segment boundaries by means of semantic audio segmentation, which is not bounded to the rhythmic restrictions of the music signals, are separated from its neighboring boundaries with a longer time interval. To improve the accuracy of segment boundaries of structural descriptions, we utilize structural change information obtained from semantic audio segmentation. This approach also solves the typical problem encountered in music structural description algorithms when dealing with songs with very short (less than 3 seconds) *intro* consisting of one or only a few strummed guitar note(s) or a short drum roll. Examples of this type of song include *Misery* and *All I've Got to Do* by The Beatles. Thus, if there appear repeated segments which include this short *intro* in the music signal, those line segments will most probably be considered as single line segments and cause inaccuracy to the boundaries of the final output structural description as shown in Figure 4.16 for the song entitled *All I've Got To Do*. The segment labeled # marks the strummed guitar sound at the beginning of the song while the dotted circles mark the ideal segment boundaries of the C-labeled segments from the output structural description.

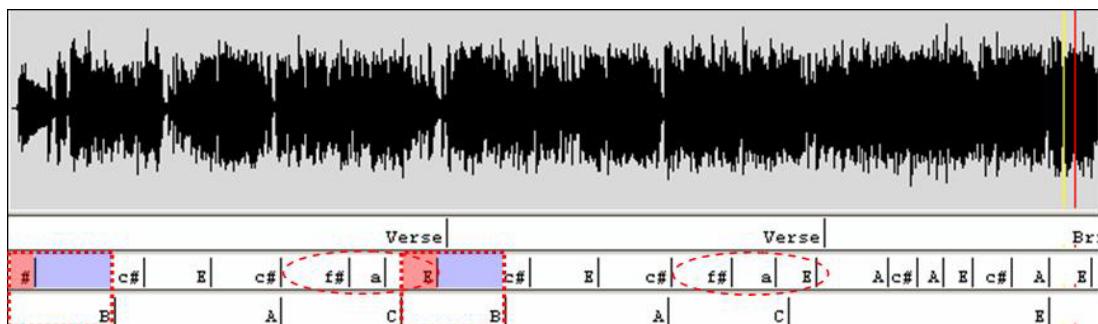


Figure 4.16. The output structural descriptions of the song entitled *All I've Got To Do*. (a) First transcription: the manually labeled ground truth result. (b) Second transcription: the chord progression of the song. (c) Third transcription: the output structural descriptions from our algorithm.

To adjust segment boundaries of the computed structural descriptor, we categorize all the repeated segments into groups according to their labels excluding those undetected segments. That is

$$\text{Group}_{A_label} = \{ \text{Segment}_1, \text{Segment}_2, \dots, \text{Segment}_m \} \quad (4.12)$$

where m is the number of repeated segments with label A. Based on the ending time of each line segment in each group, we find the nearest segment boundary from the semantic segmentation results. To be considered as a candidate group, we first make sure that each nearest semantic boundary is located at a similar direction from the line segments in the considered segment group. Figure 4.17 illustrates the alteration of line segments according to the information provided by semantic segmentation. For example, if a line segment labelled C (as shown in Figure 4.17) is considered to be lengthened, we should find its nearest semantic boundary located behind its ending time for each occurrence of C. If the considered segment group fulfils this criterion, it is then followed by considering the time distances between each line segment and its nearest segment boundary. We want to ensure that structural changes are consistent within a limited time range around the line segments. For this purpose, we check whether the absolute range of the calculated time distances (i.e. the difference between the maximum and the minimum distance) is below a threshold of 1.5 seconds. Since we do not allow overlapping segments, by changing the ending time of one segment, we also change the beginning time of the following segment. Thus, we have to examine the affected neighbouring segments for each line segment in the considered segment group. All the affected neighbouring segments should fulfill either of the following two criteria:

- (i) The affected neighbouring segment has no repetition in its music signal;
- (ii) If repetitions exist, then all the line segments with the same label as the affected neighbouring segments should also be affected neighbours. For example: If we want to extend C and this affects E, then all the occurrences of E should be found directly behind an occurrence of C.

It is noted that the first segment in the whole song is not considered in the process and thus, no alteration will be made to the first segment. This is to give some flexibility to the segment at the beginning of the song. Finally, all the ending times of the line segments of the candidate group and the beginning times of the affected neighbouring segments will be adjusted according to the average time distances computed previously between each line segment and its nearest segment boundary. The whole process is repeated based on the beginning time of each line segment in the group.

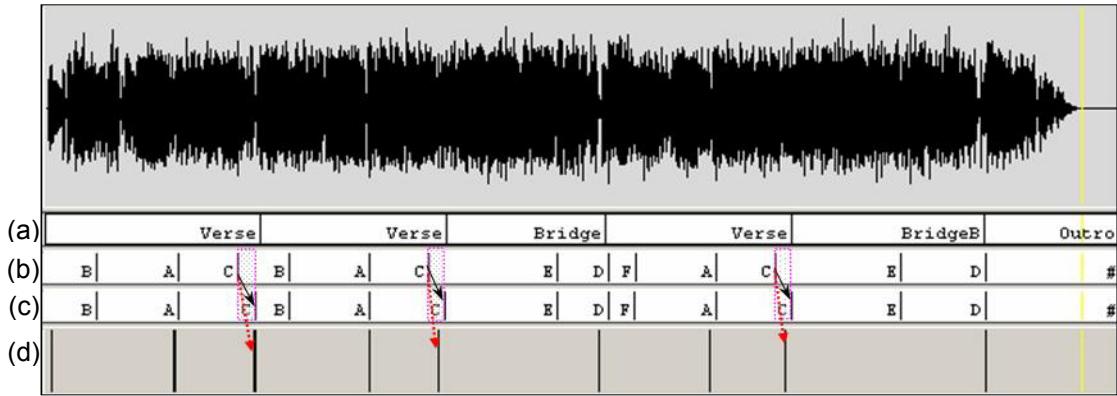


Figure 4.17. The song example entitled *All I've Got To Do* with the alteration of line segments according to the information provided by semantic segmentation. (a) First transcription: the manually labeled ground truth result. (b) Second transcription: the output structural description before boundaries adjustment procedure. (c) Third transcription: the output structural description after boundaries adjustment procedure. (d) Bottom data plot: the structural changes boundaries output from semantic audio segmentation.

4.1.8. Modulation Detection

Modulation, the process of changing from one key to another, is a very common phenomenon in music composition. Composers and song writers use modulation to give freshness to their musical compositions. In perceptual experiment, Thompson and Cuddy [Thompson92] found that both trained and untrained listeners were sensitive to changes in key and that the perceived distances of modulations corresponded well with music theoretical ideas about key distance. Thus, when analyzing music structures, one should expect to encounter and solve music modulation issues. Figure 4.18 illustrates the structural description of the song entitled *I Am Your Angel* from our proposed algorithm without any modulation detection procedure. From the ground truth, we can see that all the refrain segments appearing in this song are given the same *Refrain* label. However from the output structural descriptions from our algorithm, the first two *refrain* segments are given the label A, whilst the final two *refrain* segments are given the label B. This is because the final two *refrain* segments are modulated two semitones up from the original key of C Major to D Major. Thus, by directly comparing segment-A to segment-B, it is impossible to find any similarity within these two segments. For this reason, the algorithm without considering modulation effects fails in identifying the modulated repetitions.

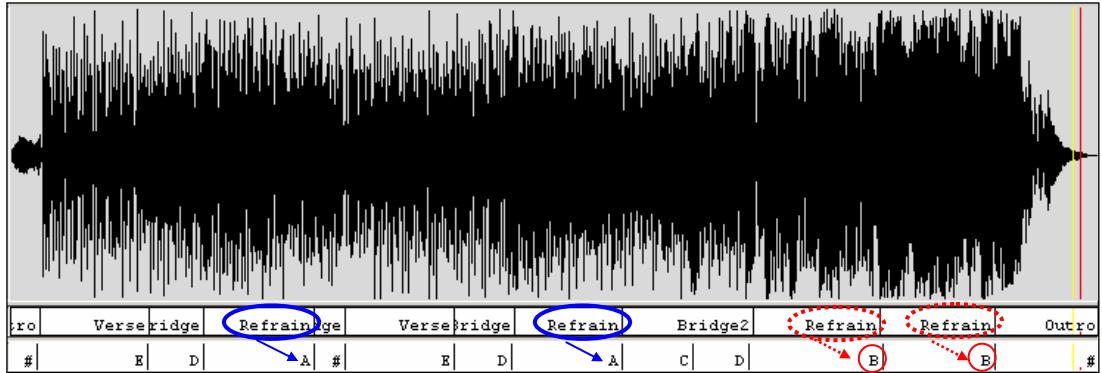


Figure 4.18. The undetected modulated “refrain” segments within the song entitled *I Am Your Angel*.
 (a) Top transcription: the manually labeled ground truth segments of music. (b) Bottom transcription:
 the detected structural descriptions from our proposed algorithm before the modulation detection
 procedure.

So far, not much study in music structural discovery has addressed the problem of modulations within a song. In our proposed approach, we are going tackle the complexity of modulation within a song by means of modifying the extracted pitch class distribution features described in Section 4.1.1. One of the advantages of octave mapping tonal descriptors is that ring shifting of the feature vectors corresponds to transposition in music perception. Since the interval resolution has been set to one-third of a semitone, ring shifting three coefficients of the features vectors resembles transposing the tonal harmonic contour by one semitone. In practical, it can be achieved by

$$v_{compared_modulated} = \begin{bmatrix} v_{compared}(:, Index_{shift}+1:36) & v_{compared}(:, 1:Index_{shift}) \end{bmatrix} \quad (4.13)$$

$$\text{where, } Index_{shift} = 3 * (r - 1) \quad (4.14)$$

and $r \in \{1, 2, 3, \dots, 12\}$ denotes the $(r-1)$ number of semitones to be modulated downwards. It is noted that when $r=1$, there is no modulation occurs at $v_{compared}$. Thus, with the newly generated music structural descriptions, we categorize all the repeated segments into groups according to their labels as shown in Equation 4.12.

Since we have no prior knowledge regarding the modulation information, we ring shift the pre-processed feature vectors, $v(n)$, as mentioned in the earlier section 4.1.2., transposing by eleven semitones downwards towards an octave. This is followed by selecting the first line segment from each group in $Group_n$, and correlating it alternatively with each of the eleven shifted feature vectors, $V_{shift-semitone}(n)$, where $shift\text{-semitone}=\{1, 2, \dots, 11\}$, is similar to the undetected repetitions recovery

procedure described in Section 4.1.5. Here, we adopt the same distance measure given by Equation 4.11. As mentioned earlier, a low distance measure indicates strong correlations within two compared windows. For detecting significant modulated repetitions, we set a constant empirical value as the upper threshold. We only consider those local minima falling below the threshold as the relevant modulated repetitions that appear in the music. Similar to the sorting and estimating candidate segments procedures explained in Section 4.1.5, we disregard those candidate segments that are not fully but partially overlapping with any of the line segments in the considered group, $Group_n$. Finally, we only consider the remaining candidate segments as the modulated segments of the compared line segments. Figure 4.19 illustrates the computed distance measures between A-Segment and the transposed (two-semitones downwards) feature vectors of the same song example shown in Figure 4.18, and the selected modulated segments. We then include those modulated segments into the group of the compared line segments and label the line segments according to the labels of the compared group. Figure 4.20 shows the output of our structural descriptions, after applying the modulation detection procedure, with the identified modulated repetitions appearing in the song example given by Figure 4.18.

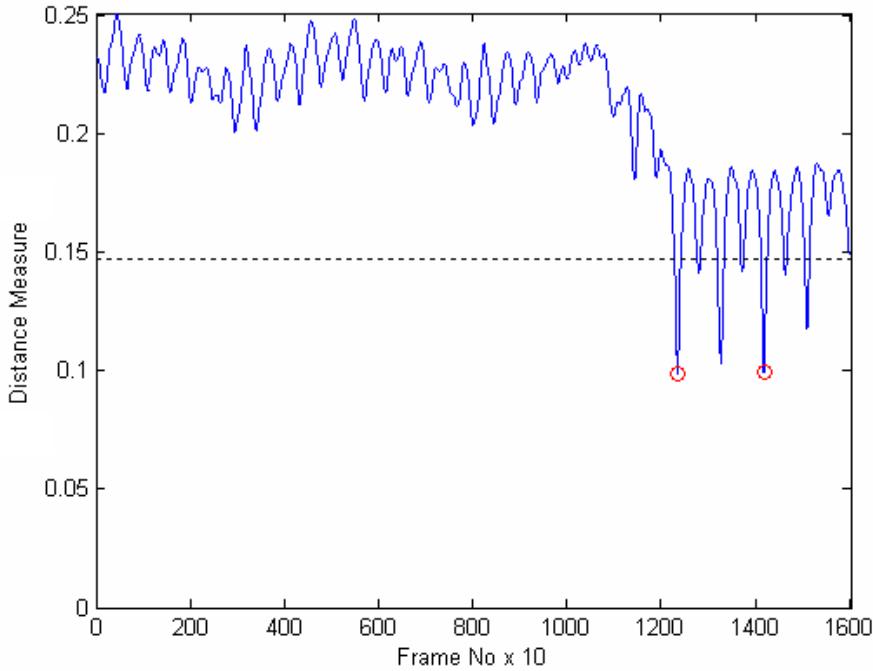


Figure 4.19. The correlation between the segment labeled A with transposed feature vectors, $V_{shift-semitone}(n)$ with the dotted line marks the predefined threshold. Circles mark the selected local minima as relevant modulated segments.

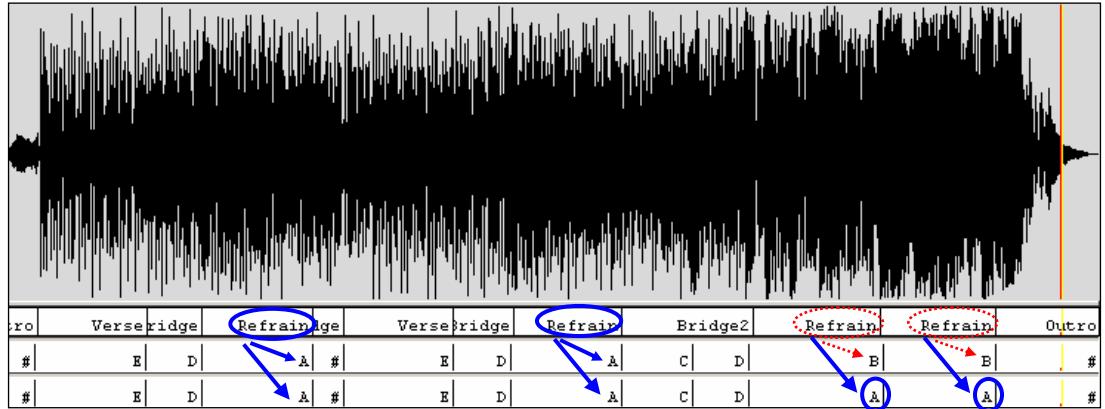


Figure 4.20. The output structural descriptions of our proposed algorithm with the same song example given by Figure 4.18. (a) Top transcription: the manually labeled ground truth result. (b) Middle transcription: music structural descriptions without the modulation detection procedure. (c) Bottom transcription: music structural descriptions with the modulation detection procedure.

4.1.9. Structural Description Inference

Finally, with the labeled line segments, we combine all the repeated labels (as shown in Figure 4.21) with a parameter, d , to restrict the maximum duration that is allowed for the integration of the repeated labels. In fact, the setting of parameter d defines the simplicity (or complexity) of the generated music structural descriptions. Figure 4.22 shows different d parameter settings with its generated structural descriptions of music from our system. As shown in Figure 4.22, with the increase of the d parameter from the most basic default computed structural description to 25 seconds, segments marked “BA” are combined to produce a single new “A” segment. Thus, the structural descriptions of the song have been modified from a detailed “BABACBABACBA” to a simplified “AACAACA” version. In evaluating the performance of our structural description algorithm, we set the d parameter as 25 sec. This is based on the assumption that structural sections in pop music (i.e. intro, verse, chorus, etc.) are less than 25 sec in length.

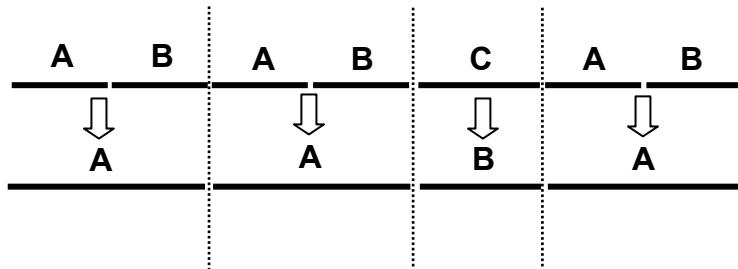


Figure 4.21. Labeling integration procedure.

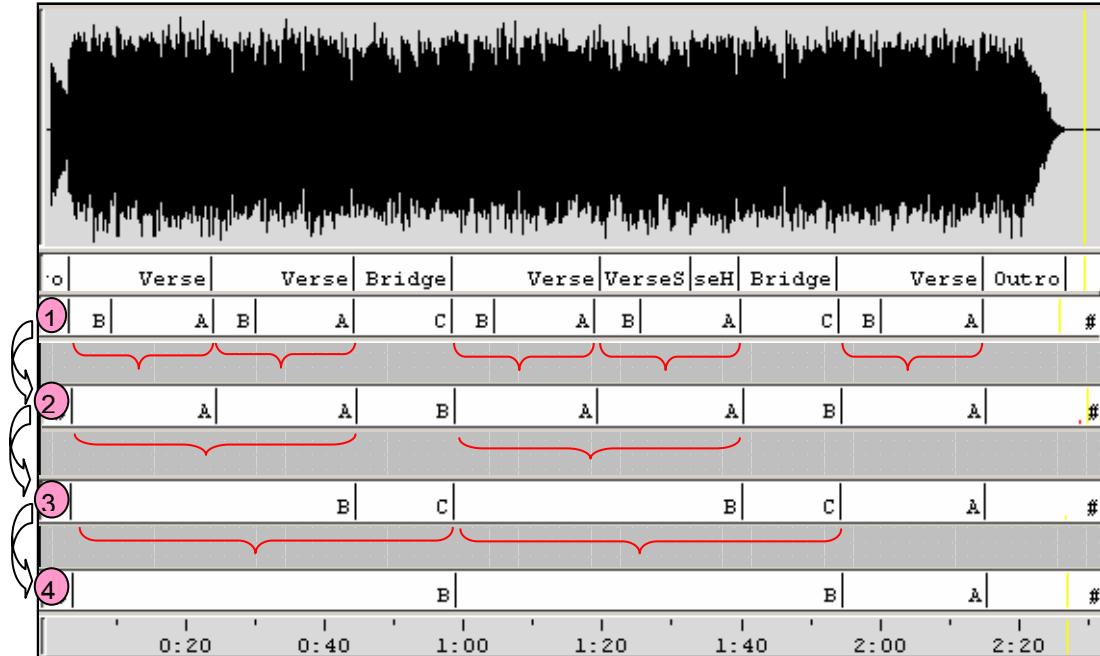


Figure 4.22. Music structural descriptions from the song entitled *A Hard Day's Night*, with various predefined d parameter settings: (1) the default structural descriptions from our algorithm. (2) 25 seconds. (3) 50 seconds. (4) 75 seconds.

4.2. Evaluation

In this section we present an evaluation of our system's performance in identifying structural descriptions of music based on different datasets. We first begin by presenting in detail our test data set and the labelling procedure. This is then followed by explaining three evaluation measures used to assess the performance of our proposed method. Then, we show the quantitative evaluation results of our system's performance for different datasets. Finally, we present our system's performance results based on applying different window sizes of morphological filtering as described in section 4.1.3.

4.2.1. Data set

In our experiments, we use three datasets. The first test set (from now onwards referred to as BeatlesMusic) consists of 56 songs from The Beatles 70s' albums, whereas the second test set (from now onwards referred to as ChaiMusic) comprises the same audio database as in [Chai05], 26 songs by The Beatles from the years 1962-1966. The third dataset (from now onwards referred to as WordPop) consists of 23 popular songs in various languages (e.g. Japanese, Mandarin, Cantonese, Indonesian, English, etc.) from different regions of the world. The selected English pop songs in WordPop include those appearing in the song list proposed in [Rentfrow03], which study the abroad

and systematic selection of music genres and personality dimensions. The purpose of using two additional datasets (i.e. ChaiMusic and WorldPop) is to evaluate the validity of our algorithm compared to other systems in existing literature. In the case of using a third test set (i.e. WorldPop) which contains songs other than The Beatles', this is to avoid having an evaluated result that biases towards The Beatles' music and also to show the applicability of our algorithm to a broad range of styles, artists, languages, and time periods within popular music.

Each song is sampled at 44.1 kHz, 16-bit mono. For evaluation purposes, we have generated a ground truth by manually labelling all the sections (i.e. intro, verse, chorus, bridge, verse, outro, etc.) of all The Beatles songs in the first two test sets (i.e. BeatlesMusic and ChaiMusic), according to the information provided by Allan W. Pollack's "Notes On" Series website on song analyses of The Beatles' twelve recording projects⁷. In the case of the third test set (i.e. WorldPop), since there exist no official song analyses available, we generated the ground truth by comparing labellings manually annotated by two advanced music conservatory students through listening to the music itself. A music composer supervised the labelling process and results.

4.2.2. Quantitative Performance

To quantitatively evaluate the segmentation performance of our algorithm, we use the standard measures in information retrieval (as explained in Chapter 3, Section 3.2.2.). We compare the obtained segment boundaries for each of the three descriptors with manually labelled ground truth results. The recall and precision are computed for various degrees of tolerance deviation (between 0.3 sec and 3.6 sec) in order to obtain a more complete picture with regards the accuracy and reliability of the segmentation results.

4.2.3. Results and Discussion

Figure 4.23 and Figure 4.24 show the evolution of precision and recall scores with respect to tolerance deviation for different pitch class distribution descriptors (i.e. HPCP, PCP, CQP and CQ-profiles) using BeatlesMusic. In both figures, we observe a significantly higher performance of HPCP compared to PCP, CQP and CQ-profiles. With a tolerance deviation of 3.6 sec, HPCP achieves a higher than 70% accuracy, and a reliability of 83%. From our segmentation results, HPCP outperforms the other tonal descriptors by as much as 10% in both precision and recall scores with 3.6 sec tolerance deviation. A t-test analysis concludes that the differences between HPCP and the rest of the used tonal descriptors are statistically significant beyond the 99% confidence level with the *p*-

⁷ The Twelve Recording Projects of the Beatles webpage: http://www.icce.rug.nl/~soundscapes/DATABASES/AWP/awp-beatles_projects.html

values < 0.01. For the case of PCP, CQP and CQ-profiles, there is no statistically significant difference in their performance on our test set.

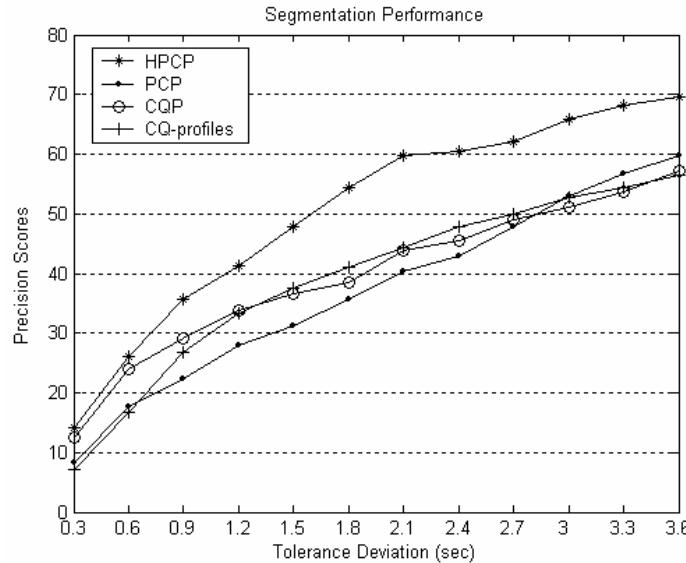


Figure 4.23. Precision measures of segmentation results (through structural analysis) with four different tonal-related descriptors using BeatlesMusic.

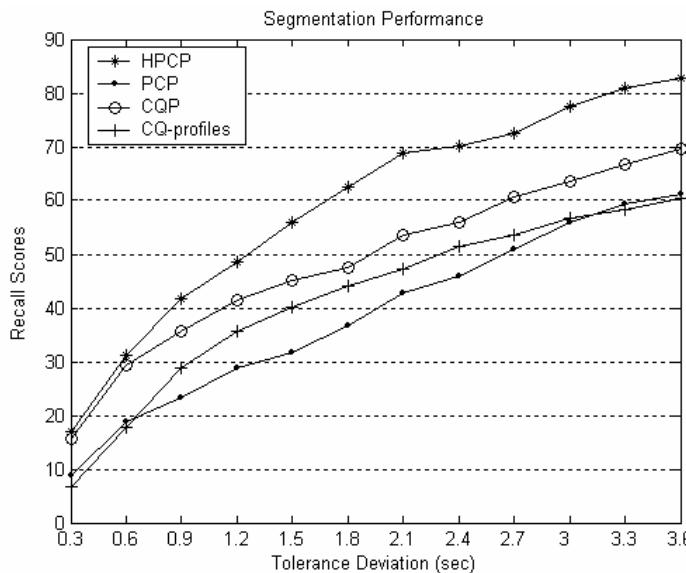


Figure 4.24. Recall measures of segmentation results (through structural analysis) with four different tonal-related descriptors using BeatlesMusic.

Comparing our proposed method with an existing structural analysis system [Chai05] by means of using a same evaluation dataset (i.e. ChaiMusic), we note a slightly better performance using our

proposed method (with the use of the HPCP tonal descriptors), throughout almost the entire evolution of recall and precision rates corresponding to the considered tolerance deviations (in seconds). With a tolerance deviation of 3.6 sec, our algorithm achieves precision and recall rates of 82% and 84%, respectively, together with an average F-measure of nearly 83%. Figure 4.25 illustrates both precision and recall scores of the HPCP using ChaiMusic with respect to the tolerance deviation.

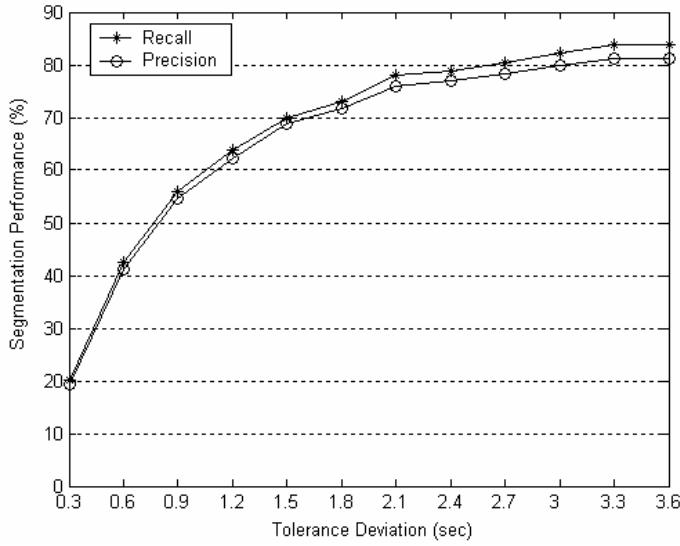


Figure 4.25. Evolution of recall and precision rates of HPCP with respect to the tolerance deviation (sec) for the different pitch class distribution features using ChaiMusic.

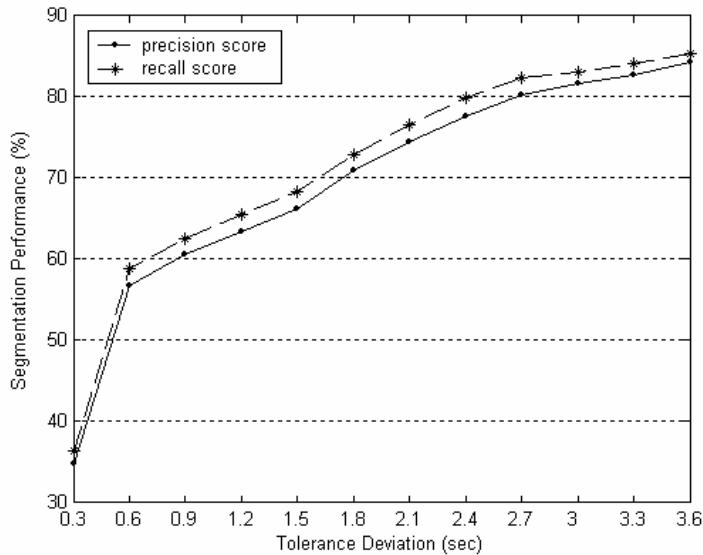


Figure 4.26. Evolution of recall and precision rates of HPCP with respect to the tolerance deviation (sec) for the different pitch class distribution features using WorldPop.

In the case of WorldMusic, which comprises of popular songs in various languages from different regions, the segmentation performance is illustrated in Figure 4.26. With a tolerance deviation of ± 3 seconds, our algorithm achieves at least 81.7% and 83.2% in its precision and recall rates respectively. Figure 4.27 shows the precision and recall scores for each song in the WorldPop with a considered tolerance deviation of ± 3 seconds. From the F-measure scores in Figure 4.27, we have observed that there are at least 7 songs (i.e. SongID-4, SongID-9, SongID-12, SongID-14, SongID-16, SongID-19 and SongID-23) which have all their structural segments correctly detected by our algorithm, with a tolerance deviation of ± 3 seconds.

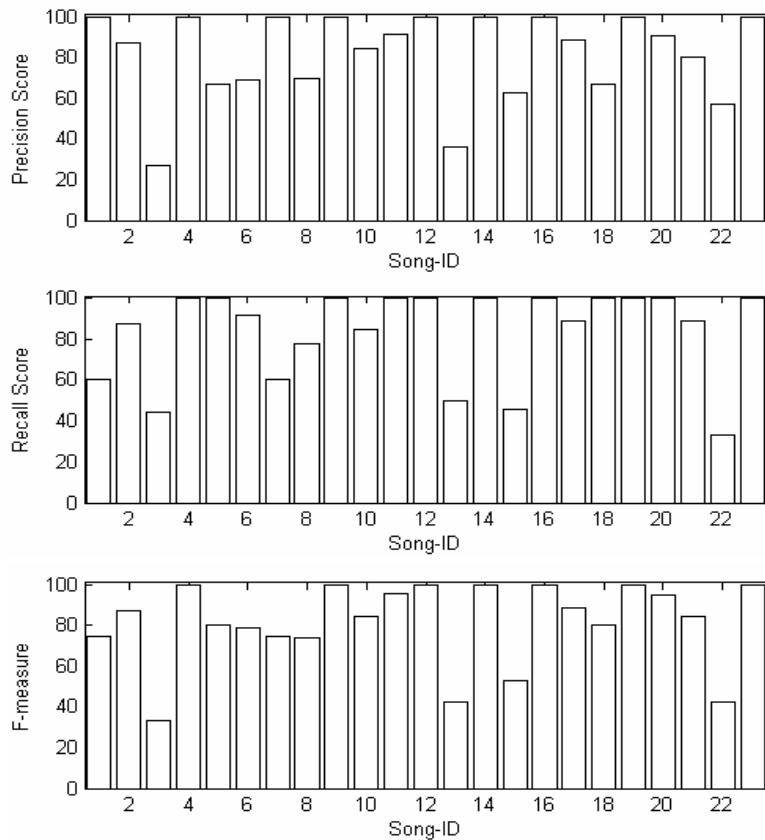


Figure 4.27. The segmentation performance (with a tolerance deviation of ± 3 seconds) on each song in WorldPop.

To compare our algorithm's performance on each of the songs in the WorldPop, we compute the average for all the F-measures obtained along the twelve considered tolerance deviations (from 0.3 sec to 3.6 sec) for each song in the test set. The distribution of the average F-measure has a mean of 69.3% and a median of 72.9%. From the median value, more than 50% of the songs have an average F-measure above 72.9%. The closeness between mean and median values show there are not many

outliers in the distribution. Figure 4.28 illustrates the average F-measures of each song in WorldPop along the various tolerance deviations. From the bar graph, we note that the best song performance in WorldPop is in the case of SongID-12. With an average F-measure of 100%, it denotes that SongID-12 has achieved 100% precision and recall rates throughout the considered tolerance deviations. In other words, all the structural segments in SongID-12 are correctly detected under 0.3 seconds tolerance deviation. Figure 4.29 shows the manually annotated ground truth result and the detected segments from our proposed structural description algorithm for SongID-12.

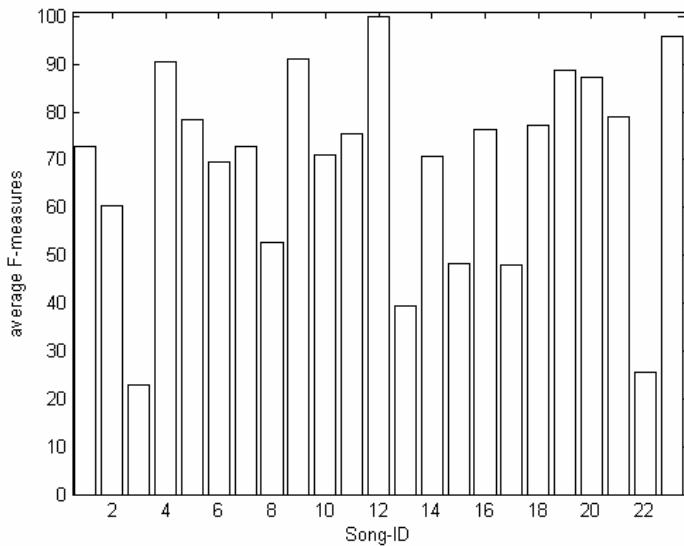


Figure 4.28. The average of total F-measures obtained from each song in WorldPop along the twelve considered tolerance deviations.

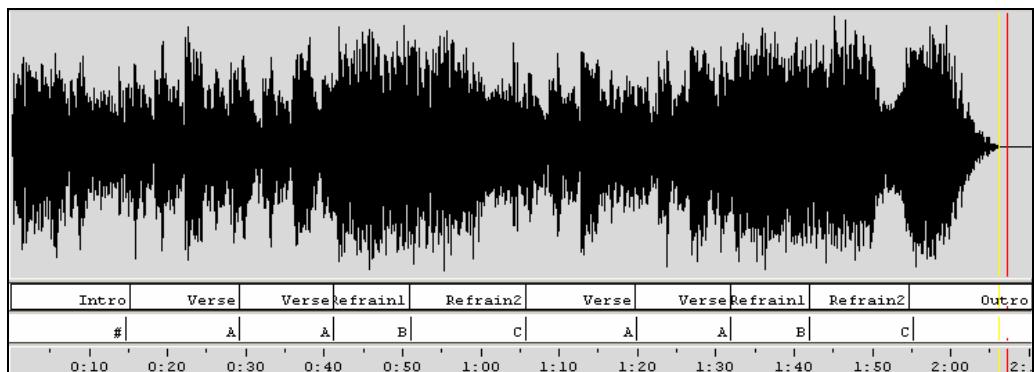


Figure 4.29. The SongID-12 entitled *I Say A Little Prayer* with two annotations: (a) Top transcription: the manually labelled ground truth result. (b) Bottom transcription: the detected segments from our proposed structural description algorithm.

Through observing the performance evolution curves generated based on each test set (i.e. BeatlesMusic, ChaiMusic, WorldPop), we find a similar abrupt increase of the segmentation performance between the first (0.3 seconds) and the second (0.6 seconds) of the considered tolerance deviation extent. In particular, WorldPop holds the highest sudden rise, as much as 22% in both precision and recall scores respectively, within this region. The sudden increase of segmentation performance within 0.3 seconds tolerance deviation to 0.6 seconds may be caused by the mixture of fricative sounds at the segment boundaries and the chosen morphological filter length. Music sections commonly start with a new phrase. Thus, the segment boundaries of music have a high possibility of containing fricative sounds incurred from the start of singing a new phrase by the singers. When there are fricative sounds that have weak tonal representations around the segment boundaries, its tonal features in the time-lag matrix will be considered as noise and will be suppressed after the binarization process. This will then incur leakage within the line segment in the time-lag matrix. When this happens, the morphological filter will remove the considered line segments, based on the chosen filter length, and thus create some errors in the detected structure boundaries. This scenario happens to be much stronger in the WorldPop data set compared to the other two test sets (i.e. BeatlesMusic and ChaiMusic), perhaps because the WorldPop database contains songs in some specific languages (e.g. Mandarin and Cantonese) which may have a higher potential of causing fricative sounds at the segment boundaries.

From the above given evaluation results, we can see that our algorithm performs quite well in discovering the structure of music by means of tonal-related features. However, we note a typical problem that occurs when using only tonal-related features in performing music structural analysis: the occurrence of true negatives in repetition identification when dealing with songs which have the same temporal evolution of chord progressions for different sections. The Beatles' song entitled *Please Mister Postman* is a typical example of different sections holding the same temporal evolution of chord progressions. Both of its "RefrainA" and "VerseA" sections are composed of identical chord progressions (i.e. A→F#m→D→E). Thus, our music structural description algorithm falsely identifies these different segments as the repeated segments in the music as shown in Figure 4.30.

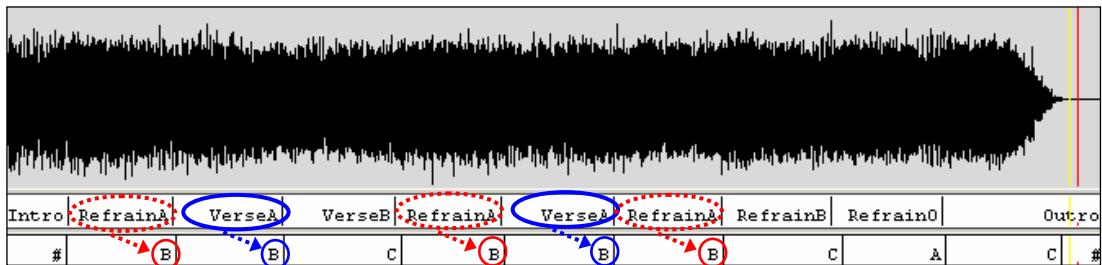


Figure 4.30. The song example entitled *Please Mister Postman*, where true negatives occur when different segments contains quite an identical temporal evolution of tonal descriptions (or chord progression in musical term). (a) Top transcription: the manually annotation ground truth result. (b) Bottom transcription: the output structural description from our algorithm.

In the following paragraphs, we are going to investigate the validity of a few hypotheses assumed by our proposed approach to music structural analysis. By means of comparing the acquired quantitative evaluation results, three aspects are examined:

- (i) The use of Euclidean distance versus cosine distance in computing distances between frame feature vectors;
- (ii) The effectiveness of coupling semantic audio segmentation into a structural analysis algorithm;
- (iii) The effectiveness of morphological filtering in music structural analysis.

Case study 1: Euclidean Distance versus Cosine Distance

To verify the validity of our observations regarding the suitability among the two distance measures (i.e. Cosine distance versus Euclidean distance) to our proposed application, we generate structural descriptions of music for all songs in BeatlesMusic based on HPCP tonal descriptors using each of these two distances. Finally, we compare the segmentation performance generated from each of the two distances to evaluate its applicability to our proposed method. Figure 4.31 shows the segmentation results generated using BeatlesMusic with respect to the tolerance deviation (in seconds) for the different distance measures: cosine distance and Euclidean distance. The plots show a clear advantage of cosine distance over Euclidean distance for generating structural descriptions of music in both precision and recall scores. Segmentation performance obtained using the cosine distance exceeded at least 6.5% in its F-measure compared to Euclidean distance. T-test analysis concludes that the differences are statistically significant beyond the 99% confidence level with the p-values<0.01. This quantitative evaluation has confirmed our preference in using the cosine distance measure to calculate the similarities between feature vectors for discovering and extracting structural descriptions from music signals using our approach.

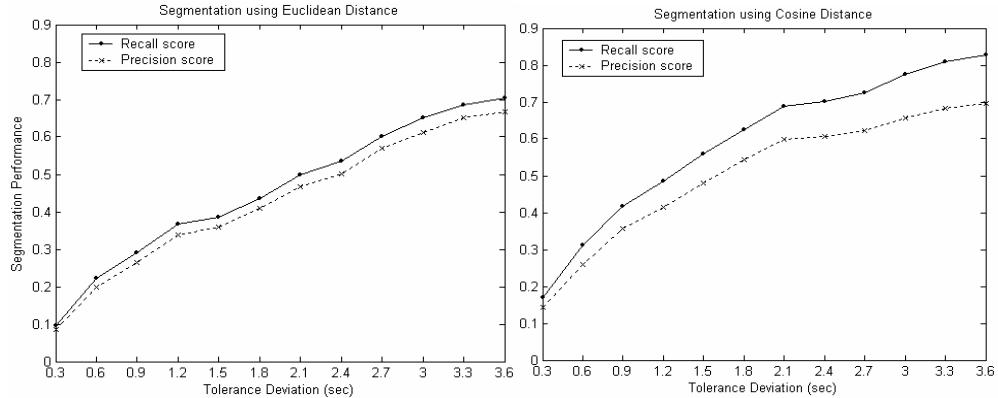


Figure 4.31. The segmentation evaluation results obtained using Euclidean distance (rightmost) versus Cosine distance (leftmost) using BeatlesMusic based on HPCP descriptors.

Case study 2: Effectiveness of Coupling Semantic Audio Segmentation

We evaluate the effectiveness of coupling semantic segmentation to our structural description system by comparing the segmentation results (with and without the use of semantic segmentation) achieved from our algorithm results. Figure 4.32 illustrates the distinct improvement in its segmentation performances with the application of semantic segmentation based on HPCP features on BeatlesMusic. The evaluation results show that our algorithm with applied semantic segmentation has improved as high as 3% in its overall effectiveness in detecting structural descriptions of music. T-test analysis concludes that the differences between the two approaches (i.e. with and without the use of semantic segmentation) are statistically significant beyond the 99% confidence level with the p-values<0.01.

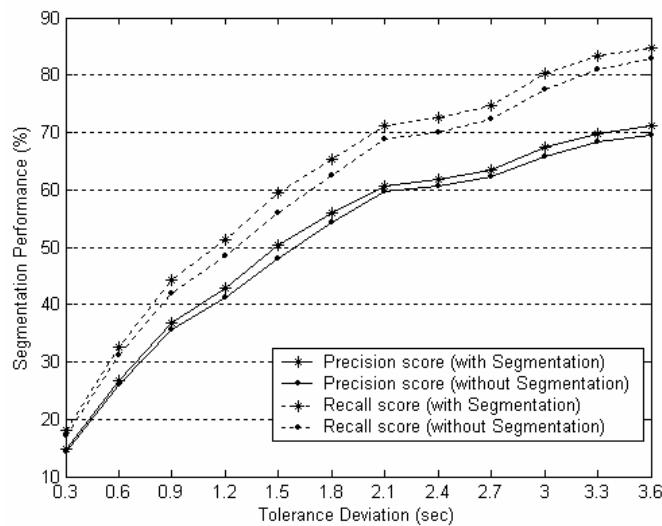


Figure 4.32. The segmentation performance with and without the application of semantic audio segmentation on our proposed structural description algorithm using BeatlesMusic.

Case study 3: Effectiveness of Morphological Filtering

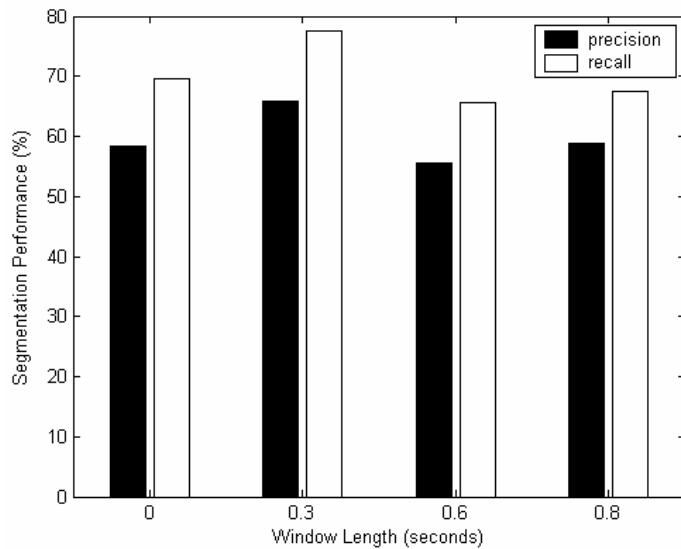


Figure 4.33. Segmentation performance of various window lengths applying to morphological filter with a tolerance deviation of ± 3 seconds using BeatlesMusic.

The performance results corresponding to different filter lengths used in the pre-processing stage has shown the effectiveness of the morphological filtering application in our system. Figure 4.33 demonstrates our system's performance corresponding to different window sizes in morphological filtering with a tolerance deviation of ± 3 seconds using BeatlesMusic. “0.0” window size in the figure denotes without the use of any morphological filter. The figure shows the dependence of the precision and recall scores on window length. The optimal performance occurs for a window size of 30 frames (approximately 0.3 seconds) with precision and recall scores of 65.8% and 77.6%, respectively with an overall F-measure of 71.2%. This surpasses the other window lengths (i.e. 0 sec, 0.6 sec, and 0.8 sec) by at least 7.0% and 8.0% for precision and recall rates respectively. The worst performance occurs for the window size of 0.6 seconds with the lowest precision rate of 55.6% and a recall rate of 65.7%. The statistical T-test concludes that the differences between 0.3 sec window size and the rest of the window lengths are statistically significant beyond the 99% confidence level with the p-values <0.01 . For the case of 0 sec (non-applied morphological filter), 0.6 sec and 0.8 sec, there is no statistically significant difference in their performance with our test set. The results demonstrate that relatively good structural descriptions can be obtained when a suitable filter length is applied to our system. As shown in Figure 3.21 in Chapter 3 regarding the histogram of the average inter-beat interval of all songs in The Beatles' database (or also BeatlesMusic in this chapter), BeatlesMusic has a mean value of 0.6 seconds with a standard deviation of 0.2 seconds. By applying morphological

filtering with a filter length approximately the same as or higher than the average inter-beat interval of songs in the database, it may lead to excessive discarding of the relevant line segments and decrease the algorithm performance. Therefore, prior knowledge regarding the beat information of the song in the processing database would be helpful for choosing an optimal filter length to be applied to the system.

4.3. Summary

In this section, we summarize the key findings from the experiments discussed in this chapter. Here, we have presented our approach towards music structural analysis by identifying and inferring repeated patterns that appear in music to generate unified high-level structural description directly from the audio signals. In addition to the obtained music structural boundaries information, similar with the one provided by semantic audio segmentation explained in the previous chapter, (dis)similar sections in music are identified and tagged with explicit labeling. We have investigated using different tonal-related features to discover repeated patterns appearing in music for later structural description generation. By integrating semantic audio segmentation to our music structural description system, modest improvement to both the accuracy and the reliability of our system is achieved. In addition, we have also built a music structural description system that detects modulations within a song. Experiments were conducted to evaluate the performance of our proposed approach for polyphonic audio recordings of popular music. Additionally, we also studied the effectiveness of morphological filters for pre-processing the signal prior to the identification process through the use of various window sizes.

In the next chapter, we will present our new approach in identifying representative excerpts from the audio signal based on the generated music structural descriptions. Additionally, subjective evaluation is conducted by music listeners through an online listening test that examines the quality of the extracted segments from various approaches based on human perceptions.

Chapter 5

Identifying Representative Audio Excerpts from Music Audio

In the previous chapter, we presented our system for discovering and extracting music structural descriptions from audio signals. In this chapter, we continue our research study with the identification of representative audio excerpts from music signals based on the structural information generated using our system described in the previous chapter. In other words, our aim is to generate an audio excerpt that captures the retrieval cue or the gist of the music input signal. In line with this, we study the significance of various approaches related to this context through subjective evaluation based on human perceptions. In particular, three different approaches are investigated. The first approach pursues the widely used manner by online music stores (e.g. Amazon, iTunes, etc.) in previewing music, where the first 30 seconds of the song is selected to represent any piece of music. The second approach emphasizes the significance of the most repetitive excerpts in the music. The third approach considers all repetitions as equivalent. Instead, high similarity between a repetitive segment and the entire song is used as the primary criteria to select the best suitable audio excerpt to represent a piece of music. To make this study possible, we have set up an online listening test. Music listeners were invited to participate in the online listening test to examine the quality of the extracted segments using the various approaches.

So far, a few studies related to evaluating various strategies for music summarization have been conducted [Chai05] [Logan01]. Logan and Chu [Logan01] conducted user tests to evaluate the quality of music summaries obtained from selecting a fixed 10 seconds segment among the frames with the most frequent label assigned by two bottom-up clustering techniques (i.e. clustering approach versus

HMM approach). Chai [Chai05] also investigated various strategies for music summarization (i.e. random; beginning of the second repeated section; transition between the most repeated section and the second repeated section; transition between the second repeated section and the most repeated section). From Chai's compared summarization strategies, the author reported a consistently high performance for cases that extract music summaries based on the beginning of the most repeated section over the other strategies

5.1. Audio Excerpt Identification and Extraction

Given that one of our aims in this chapter is to identify a representative excerpt from music signals, one may ask, "what is the criterion required for a music section to be acknowledged as a representative excerpt the whole piece of music?" As mentioned in Chapter 1, the repetition, transformation, simplification, elaboration and evolution of music structures create the uniqueness of the music itself. Hence, many research in this area have assumed that the most representative sections of music are frequently repeated within the song. In fact, this has been the most adopted assumption for generating the most representative excerpt or thumbnail of any music in audio research [Chai05] [Logan01]. We acknowledge the significance of repetitiveness of music in human perception and cognition. However we hypothesize that perhaps there may exist some other significant factors (e.g. first 30 seconds of the piece as they are the most memorable, etc.) and some of them will be explored in the next sections. In this study, we consider three approaches in extracting a representative excerpt of a music signals. Our implemented system extracts a short excerpt (with a fixed length of 30 seconds) from each song based on these three approaches, to serve as the representative excerpt of the music signal. We do not allow any differences in duration between the extracted audio excerpts, to avoid having the subjects' preferring excerpts with longer durations. The following sections explain in detail the three investigated approaches.

5.1.1. First-30-seconds

In identifying a representative excerpt from a music signal, we assume that such audio excerpts can also serve as a retrieval cue of the music wherein when one listens to the particular audio excerpt, he or she would be able to tell if this is the music that he or she is looking for or if it is interesting. Since most online music stores utilize the 30 starting seconds of the audio signal for music previewing, we include this criterion within our study of comparing different approaches used in identifying representative excerpts of music signals.

5.1.2. Most-repetitive

In our second and third approaches, we exploit the structural information generated from our music structural description system to extract the significant representative audio excerpts from music signals. As explained in Chapter 4, our music structural description system produces structural transcriptions with the use of labeling (i.e. A, B, C, etc.) and time-stamping to mark (dis) similar sections that appear in the music signal (i.e. verse, chorus, bridge, etc.). Based on generated structural information, we first categorize all the repeated segments into groups according to their labels, excluding those undetected segments. That is

$$\text{Group}_A_label = \{Segment_1, Segment_2, \dots, Segment_m\} \quad (5.1)$$

where m is the number of repeated segments with label A. In each repetition group, we sort the repeated line segment in ascending order based on their time information. As mentioned above, the second approach gives more priority to the repetitiveness of an audio segment. Since repetitiveness of each repetition group is defined by the number of line segments it encompasses, we begin the identification process with calculating the number of line segments included in each repetition group (as described in Chapter 4, section 4.1.7). Repetition groups with the highest number of line segments are selected as group candidates for having the possibility of comprising the most representative segments. The idea of extracting representative excerpts comes from applications such as music recognition, audio browsing, audio thumbnailing and so forth. Thus, for selecting representative excerpts of music for such applications, we would prefer to extract audio segments which appear to be the most original among the rest of the repetitions in the same group. Here, we assume that such original repetitions normally appear first in a repetition group. Thus, we extract 30 seconds of audio from the starting time of the first repeated segment in the most repeated group (which will be then considered to be the most representative excerpt of the music piece).

5.1.3. Segment-to-Song

The third approach explores the potential of the segments in capturing the specific features of the representative excerpt of the music. Since the third approach focuses more on content descriptions of the unique characteristics of music sections, we disregard the repetitiveness of audio segments but consider all repetition groups as equally important in making group candidates selection. Contrary to the first approach, we place importance on the potential of the segment to grasp the specific properties of the entire song. Here, we truncate songs into several segments based on the time information subsumed in the song's generated structural descriptions. The audio features are then extracted on a frame-by-frame basis. We extract audio features from MPEG-7 descriptors from each repeated

segment. Table 5.1 lists the above-mentioned audio features categorized into three different groups. We group all frames included in each repeated segment and compute average values for each feature. We then use a Manhattan distance [Tzanetakis04], defined as $d(f, f') = |v_1 - v'_1| + \dots + |v_m - v'_m|$, to access the distance between each repeated segment to its entire song. Figure 5.1 illustrates the procedure of distance computation between each repeated segment to its full-length song.

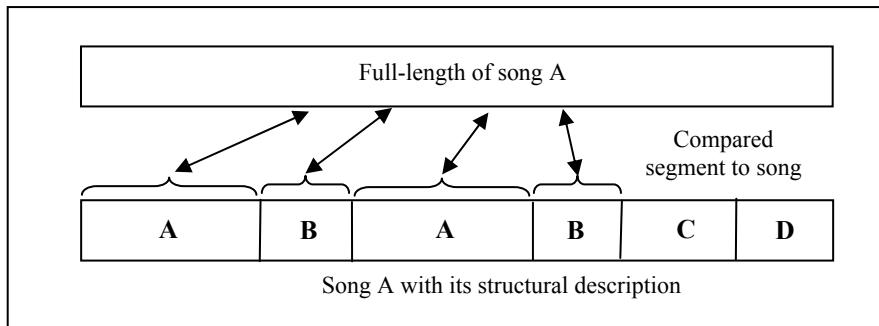


Figure 5.1. Segment-to-song distance computation.

Features Group	Audio Features
Spectral & Temporal features	Energy; Intensity; Loudness; High-frequency coefficient; Low-frequency energy relation; Maximum magnitude frequency; Spectral rolloff; Spectral centroid; Spectral flatness; Spectral decrease (the amount of decreasing of the spectral amplitude); Spectral kurtosis; Spectral skewness; Spectral spread; Strong peak; Zero crossing rate;
Bark band	Bark Band
MFCC	Mel-Frequency Cepstral Coefficients

Table 5.1 Features grouping extracted from audio segments.

No	Artist	Song Title
1	The Beatles	It Won't Be Long
2	The Beatles	No Reply
3	The Beatles	All My Loving
4	The Beatles	If I Fell
5	The Beatles	Can't Buy Me Love
6	The Beatles	Please Mister Postman
7	The Beatles	Eight Days A Week
8	The Beatles	Things We Said Today
9	The Beatles	Do You Want To Know A Secret
10	Aretha Franklin	Chain Of Fools
11	Rolling Stones	Brown Sugar
12	Air Supply	I Can't Wait Forever
13	R. Kelly & Celine Dion	I Am Your Angel
14	Whitney Houston & Mariah Carey	I Believe In You And Me
15	Destiny's Child	Independent Women
16	Van Halen	Jump
17	Mike Francis	Room In Your Heart
18	N'sync	Bye Bye Bye

Table 5.2. Eighteen music pieces used in the online subjective evaluation.

Finally, of all segments belonging to one song, we select the one with the smallest distance d to the entire song. We then extract 30 seconds from the beginning of the segment to represent the whole song itself.

5.2. Evaluation

Since the ultimate users of music is the audience, we created an online listening test to obtain a subjective evaluation of the extracted audio excerpts using the different approaches based on judgment by human listeners. In our experiment, we created an audio database consisting of 9 popular songs from The Beatles' and another 9 popular songs from other artists or groups (Table 5.2). Song titles of the pieces and artists names were not provided to the subjects during the experiments.

5.2.1. Subjects

Subjects were invited through posted announcements at the FreeSound forum and emails to a few different groups such as the MTG mailing list and the Summer School on Sound and Music Computing 2006 mailing list.

5.2.2. Datasets

For all the 18 pieces, three music excerpts were extracted using the three different approaches. Thus, there were 54 song excerpts in total in our listening test. In each listening test, a list of 9 song excerpts

were chosen for each subject evaluation. This specific number was chosen based on the consideration that each subject should take approximately 30 minutes to finish the listening test. Considering that our listening test relied on voluntary participation, we had to stay within an acceptable time spent by our subjects. In this case, we set a time limit of 30 minutes. For every 6 consecutive listening tests, the subjects would evaluate the same 9 song excerpts as the first listening test. We understand that the more song pieces included for evaluation, the more representative the obtained results from the listening experiment. However considering that there would be a limitation in the number of participants in the listening test, we only included 18 music pieces for this experiment, so that each song excerpt would be evaluated by more than one subject. All the excerpts together with the original songs were converted to MP3 format (sampling rate of 22 kHz, Mono-channel, 56 kbps). For each excerpt, four questions were asked to seek information regarding the following:

- Familiarity with the sound excerpt;
- Amount of effort required to recall the song;
- Subjects' satisfaction regarding the use of the presented sound excerpt to represent the entire piece of music.

It is noted that throughout the online survey, the information regarding the approaches used in identifying representative excerpts from music signals was not disclosed to the subjects.

5.2.3. Web Interface and Listening Test Procedures

The web interface used in this online listening experiment was based on that used by [Sandvold05] for a different purpose, and was later also adapted by [Streich06] for music complexity judgment. In the online listening test each subject was given 9 audio excerpts for evaluation, taking an estimated time of approximately 30 minutes to complete the whole test. As three excerpts were extracted from each piece in the test data using different approaches, there was a possibility that the subject might be asked to evaluate different audio excerpts of the same songs several times. Below are the descriptions of our online listening test following its proceeding order.

1. Introduction: The online listening test begins with an introduction page together with a link to test the installed web browser audio plug-in as shown in Figure 5.2. The subjects were given options of either installing a plug-in or downloading the mp3 files and listening through their own mp3-player application.



Figure 5.2. Introduction page of our online listening test.

2. Subject registration: Subjects were then prompted to a registration page where a few personal questions regarding gender, age, musical background, familiarity with popular music, familiarity with The Beatles' music as well as how much subjects actually liked their music before proceed to the evaluation pages. Figure 5.3 illustrates one of the subject registration pages.



Figure 5.3. Subject registration page.

3. Before getting started with the evaluation process, subjects were sent to the introduction page of the evaluation site. This page describes the questions asked of the subjects.
4. Audio excerpts evaluation: Each page presented one song excerpt from the test database to be rated by the subjects. For each audio excerpt, four questions were asked in the following order:

Question 1: Have you heard this song before?

Question 2: How much effort do you need to recall the whole piece of music if you have heard this song before?

Question 3: Please select one of the labels below which you think is the most likely title of the song.

Question 4: Please listen to the above full-length as well as 30 second excerpts of the song.

How would you judge the quality of the short excerpt used as a music summary of the full-length song?

Four choices of song titles were generated based on the song lyrics that appeared in all the extracted song excerpts from the same song. This was to increase the difficulty level in defining the song titles through listening to the lyrics that appeared in the presented song excerpts. Thumbnail rating was used. Figure 5.4 presents the evaluation page. As shown in Figure 5.4, if subjects needed

help in answering a particular question, they could click on the help link, which would redirect them to the relevant help page. This page gave some hints to subjects on how to rate the song excerpt according to the question. Figure 5.5 and Figure 5.6 illustrate the help pages for Question 2 and Question 4 listed above. In addition, subjects were given an opportunity to explain why they considered an excerpt to be a “poor” or “bad” summary of the song.

Note: Please note that in this listening experiment, song with different summaries can be presented several times.

Song 1 of 9
No plug-in? [Click here for playback!](#)

a. Have you heard this song before?

Yes.
 No.

b. How much effort do you need to recall the whole piece of music? - [\(Help\)](#)

No effort at all.
(It comes to my mind immediately)
 A little.
(It needs to take some time to recall)
 A lot.
(It asked me a lot of effort to recognize it)
 I can't.

c. Please select one of the labels below which you think is most likely the title of the song? You are allowed to replay the audio example as many times as you wish.

Getting Closer
 Alone
 Room In Your Heart
 Tonight

d. Please listen to the above full-length and 30 seconds excerpt of the song. How would you judge the quality of the short excerpt used as music summary of the full-length song? - [\(Help\)](#)

[Whole song](#)
[Song excerpt](#)

Excellent.
 Good.
 Fair.
 Poor. Please explain why you choose POOR grading.
 Bad. Please explain why you choose BAD grading.

[Continue](#)

Figure 5.4. The evaluation page.

Judging "Music Cue"

Music cue facilitates in recalling or recognizing songs that we have heard before. Thus, when listening to the played audio excerpts, please try to consider the following questions:

- Do you need to take some time to say "I have heard this song before!"?
- Does the original song come to your mind immediately?
- Does it demand a high effort to recall other parts or segments of the song?

*Note: If your previous answer in (a) is "NO", the answer here should be "I CAN'T".

[Contact](#)



www.semanticaudio.org

Figure 5.5. Help page for Question-2.

Judging Music Summaries

Music summaries facilitates in browsing large quantity of audio pieces without having to listen to the whole piece. Thus, when listening to the played audio excerpts, please try to imagine the following context:

You are browsing or using a music downloading or recommendation service to purchase some music for yourself. The music provider offers you a gist of the songs with the length of 30 seconds to speed up your browsing process by not having to listen to the full-length of the recommended songs. So...

- Does this 30 seconds excerpt manage to capture the most characteristics segment of the full-length song?
- How well is it to use this 30 seconds excerpt to give an overview of the full-length song?

[Contact](#)



www.semanticaudio.org

Figure 5.6. Help page for Question-4.

5. After evaluating all the 9 audio excerpts, subjects were redirected to the feedback page where they were asked their opinion as to what would be the best way to generate a good music summary, identifying factors that they considered important in rating their satisfaction with the presented audio excerpts. Figure 5.7 shows the feedback page of the listening test.

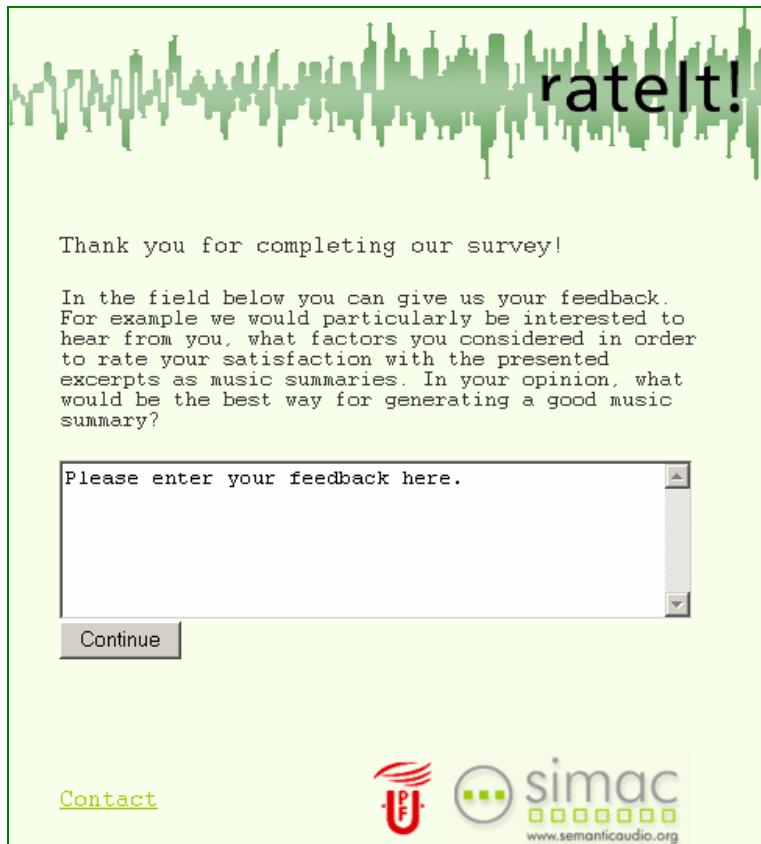


Figure 5.7. Feedback page.

5.3. Observations and Results

A total of 44 subjects participated in the listening test. Figure 5.8 illustrated the age histogram of the participants. The overall participants had a mean age of 33 and a standard deviation of 8.7. The majority of the participants were within the age of 24 to 35 years old. Among the rest of the participants, there were 9 participants who fell below the age of 26 years old; 8 participants within the age range of 36 to 45; 2 participants who were over 45 years old but less than 60 years old and finally 1 participant who was over 60 years old. Among the participants, 7 subjects had no musical background, 12 subjects had a basic musical background, while 17 and 8 subjects respectively had

advanced and professional musical backgrounds. Figure 5.9 illustrates the evaluated song excerpts histogram by different musical backgrounds from the subjects.

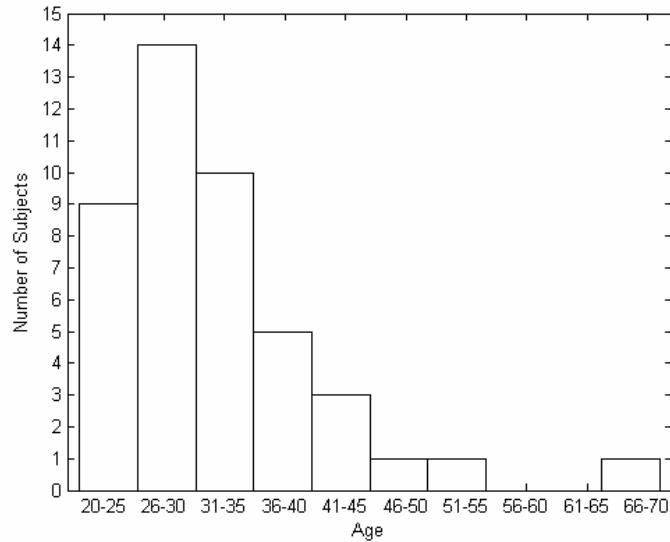


Figure 5.8. Subjects' age histogram.

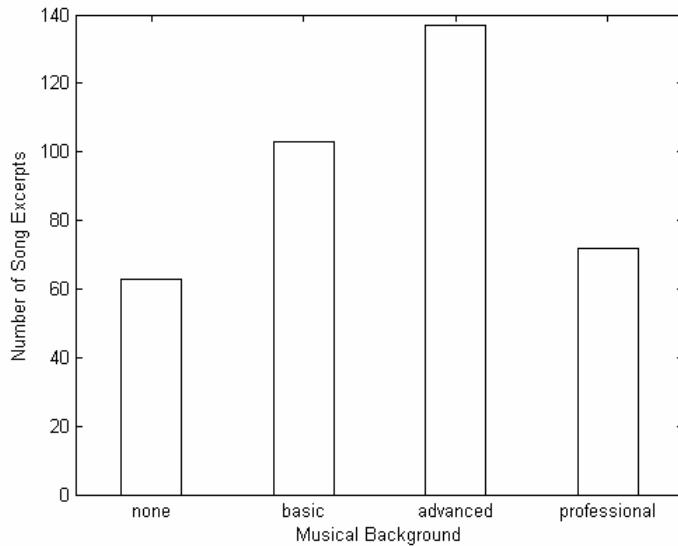


Figure 5.9. The evaluated song excerpts histogram according to subjects' musical background.

Objective Evaluation

For evaluating the efficiency of various approaches in extracting song excerpts which contain the song titles, we generated a quantitative estimation by means of listening to the lyrics of all the song excerpts and identifying the occurrence of the song titles in those excerpts extracted based on each

approach. Table 5.3 illustrates the quantitative evaluation results of song titles included in the song excerpts corresponding to the use of different excerpt identification approaches.

Approach	First-30-seconds	Most-repetitive	Segment-to-song
Generated audio excerpt with enclosed song titles	55.6%	72.2%	83.3%

Table 5.3. Objective evaluation results of song titles included in the excerpts generated using different approaches.

To obtain an average summary quality score for each approach, we recorded the label into the numerical ordinal scale corresponding to the {‘bad’, ‘poor’, ‘fair’, ‘good’, ‘excellent’} remarks used in the web survey. Figure 5.10 shows the overall ratings collected from the listening test. From the obtained results, we note that the segment-to-song approach for representative excerpt identification achieved the highest score in identifying the song titles. It surpassed by as much as 5.7% and 10.3% compared to the most-repetitive and the first-30-seconds approaches in naming the title of the songs by all the subjects. The acquired results are consistent with the objective evaluation shown in Table 5.3. For representing a piece of music, the most-repetitive approach achieved slightly better comments from the participants compared to both the segment-to-song and the first-30-seconds approaches. Summary quality scores of {0-0.2, 0.2-0.4, 0.4-0.6, 0.6-0.8, 0.8-1.0} appear in the bar chart corresponding to the {‘bad’, ‘poor’, ‘fair’, ‘good’, ‘excellent’} remarks used in the web survey.

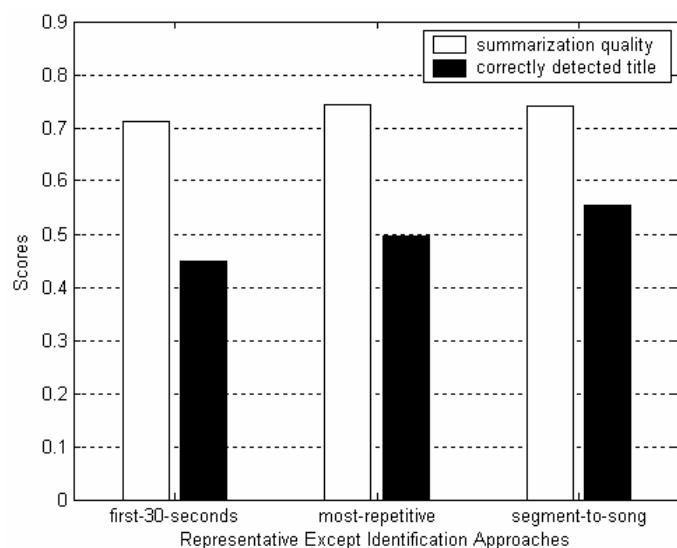


Figure 5.10. The overall ratings of the subjective evaluation.

Musical Background

By categorizing subjects based on their musical backgrounds, we observe two opinion patterns appearing in their preferences for approaches used in identifying a representative excerpt from music signals. Figure 5.11 shows the overall summary quality ratings for each approach according to subjects' musical backgrounds. As shown in the bar chart, subjects with professional and advanced levels of musical background prefer the segment-to-song approach over the most-repetitive approach in identifying representative excerpts from music signals. In contrast, subjects with basic and no musical backgrounds prefer the most-repetitive approach over the segment-to-song approach. Overall, using the first-30-seconds approach to represent a piece of music is the least preferred among all subjects regardless of musical background.

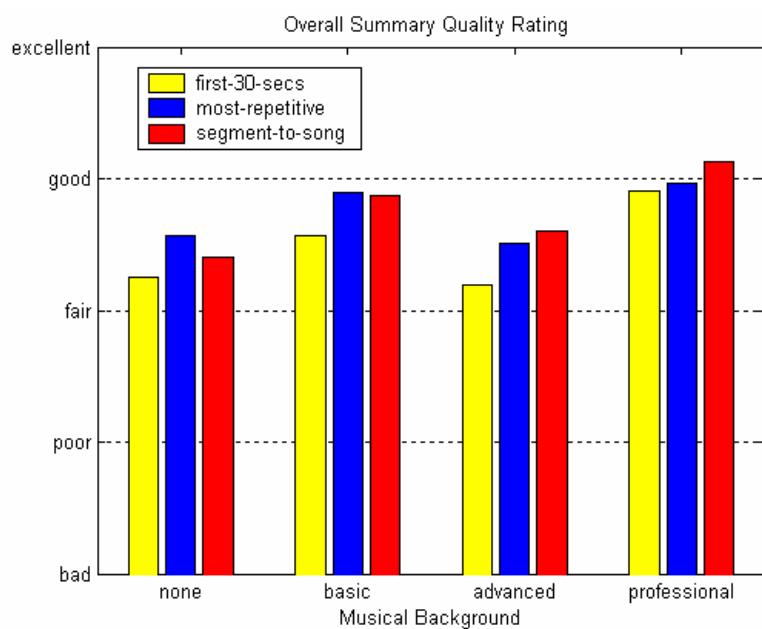


Figure 5.11. The overall summary quality ratings for each approach used in identifying representative excerpts from music signals according to subjects' musical backgrounds.

Song Familiarity

Figure 5.12 shows the overall song titles identification accuracy and summary quality obtained based on the subjects' familiarity with the song excerpts. Logically, those that have heard a song before would be better in naming the song title compared to those that are not familiar with the presented songs as the presented subjective evaluation results show. Overall, subjects who were familiar with the presented songs were at least 30% better in identifying the correct song titles than those who were not. In addition, the bottom bar chart shows that subjects, who were familiar with the songs, preferred the segment-to-song approach for identifying representative excerpts from music signals. This is

different from those who had not heard the songs before. In this latter case, they preferred the most-repetitive approach. A t-test analysis concludes that the differences of the preferences based on song familiarity for first-30-seconds approach and segment-to-song approach are statistically significant beyond the 95% confidence level with the p -values <0.05 and p -values <0.01 , respectively. However there is no significant difference for the most-repetitive approach based on song familiarity.

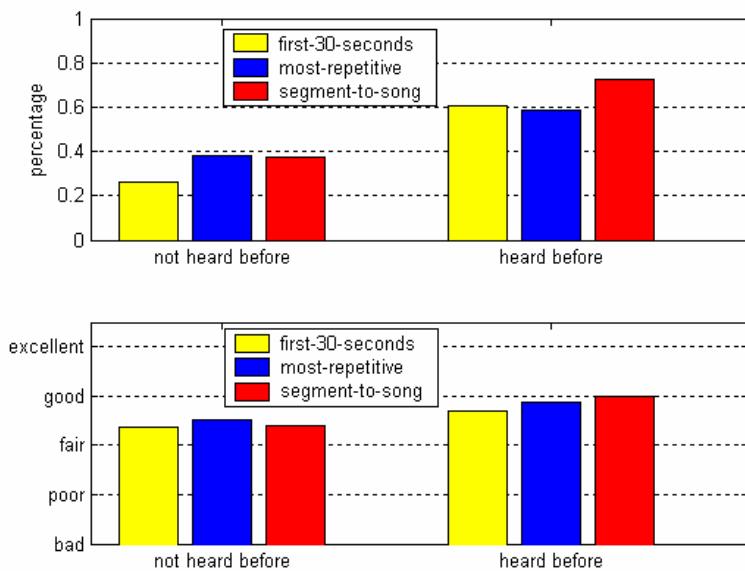


Figure 5.12. (Top) The song titles identification accuracy and (below) overall song summary quality obtained based on subjects' familiarity to the presented song excerpts.

Various Approaches vs. Recall Effort

Figure 5.13 illustrates the amount of effort required to recall a piece of music that was heard before, based on song familiarity and representative excerpts identification approaches. Here, we only consider the answers of those subjects who are familiar with and have heard the music before. Our results indicate that for these subjects, the most-repetitive approach appears to require slightly less effort in recalling the music than the first-30-seconds method. However the difference between these two is statistically not significant. This could be due to the primacy effect in long term memory that makes it easier to remember the repeated elements. Alternatively, the first-appearing elements in a list or the first seconds of a song have probably been less affected by distractions from subsequent excerpts. These results indicate that the first-30-seconds and the most-repetitive approaches play an important role in recalling a piece of music under different circumstances. In this context, the segment-to-song approach seems less practical.

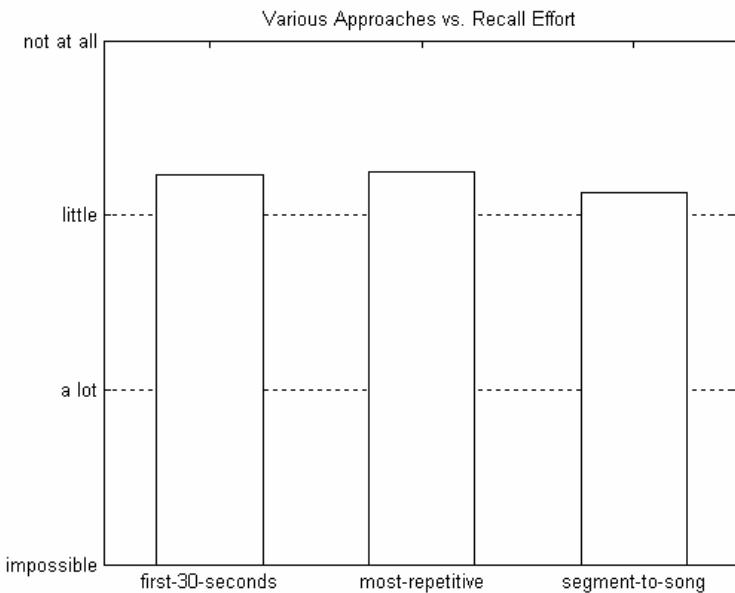


Figure 5.13. The recall effort required on each approach based on songs familiarity (Description: x-axis represents various approaches and y-axis denotes recall effort).

5.4. Summary

In this chapter, we have presented an online subject listening test to evaluate the significance of various approaches used in identifying representative excerpts from music signals. We have explored a novel possible approach, the segment-to-song approach, to detect significant representative excerpts of the music. Both subjective and objective evaluation results for song title identification are consistent and have shown that our proposed segment-to-song approach is better in capturing the song titles over other evaluated approaches (i.e. first-30-seconds approach and most-repetitive approach). Even though the overall summary quality of our proposed approach did not surpass the other approaches, it was preferred by subjects with stronger musical backgrounds. For the next chapter, we move towards music retrieval issues based on music structural descriptions. Therein, we present our approach in extracting useful representative audio excerpts or summaries from audio signals, based on music structural descriptions, for retrieving different versions of the same song in music collections. The following chapter includes an objective comparison and evaluation of the performance of our proposed identification method with the use of 90 mixture versions from 30 different songs from popular music.

Chapter 6

Structural Descriptions for Song Version Identification in Music Collections

In Chapter 4, we described a method for automatically generating music structural descriptions from music audio signals. A system that can provide high-level descriptions of music signals should be feasible to be exploited in other tasks, such as music data mining and music retrieval, besides direct provision of musical information. We hypothesize that the prior knowledge about the structural descriptions of the music would give a better grasp of the musical content and contribute to effective retrieval of large amounts of digital audio data. In this chapter, we present our approach towards retrieving different versions of the same song by means of exploiting the representative audio excerpts or summaries from audio signals, based on its music structural descriptions.

We have implemented our own system to perform the task of identification of song versions with retrieval based on prior music structural information obtained from our automatic structural analysis system described in Chapter 4. After applying particular criteria in segment selection, which will be explained in detail below, the system extracts fixed-length short summaries or segments from the full-length song for further identification processing. Finally, to evaluate the applicability of our proposed method, we compare the retrieval performance obtained using our approach with the one obtained using the whole-song approach proposed by [Gómez06b], which focuses on analyzing the similarity of tonal features in identifying different versions of the same piece, using a similar test set.

6.1. Short Summary Approach

Current literature in identifying representative excerpts of music audio mainly focus on music summarization and thumbnailing. So far, from our literature survey, there is no publication or report on using audio derived structural description to identify different versions of the same song in music collections for music retrieval or song recommendation purposes. Moreover, most literature in identifying representation excerpts of music pay great attention to the significance of repetitions in music. In the exiting literature [Logan00] [Bartsch01], the most repetitive segments are considered as the most significant excerpts to represent a piece of music. Considering its application context in version retrieval, we explore the potential of some other factors that could be useful to retrieve songs with its different versions. In our short-summary approach, we investigate two ways of identifying representative excerpts of music for version identification purposes with the use of Harmonic Pitch Class Profiles (HPCP) features [Gómez06a]. Following the commonly used criteria, the first approach emphasizes more on the significance of the most repetitive excerpts in music. The second approach considers all repetitions as equivalent. Thus, the total duration of all identical repeated patterns are taken as the highest priority factor in selecting the best suitable audio excerpts to represent a piece of music.

Based on the structural analysis results obtained via previous steps, we categorize all the repeated segments into groups according to their labels. That is

$$\text{Group}_{A_label} = \left\{ \text{Segment}_1, \text{Segment}_2, \dots, \text{Segment}_m \right\} \quad (6.1)$$

where m is the number of repeated segments with label A.

6.1.1. Repetitiveness Emphasized

Since the number of elements in a group, Group_{A_label} , denotes the occurrence frequency of label A in the music, the group with the highest m value marks the most repetitive segment group of the music. Thus, the first approach, which emphasizes the significance of the most repetitive excerpts in music, selects the group with the highest m value and extracts a fixed duration, l seconds, from the starting-time information of the group's first segment.

6.1.2. Repetitiveness-Equivalence Emphasized

In the second approach, we hypothesize that different versions of the same piece of music may vary in its musical structure. For example, the most repetitive segments of the query song may not appear to be the most repetitive segments in its song versions. Considering this issue, we generate two short summaries or segments from a song in order to overcome instances which have variances in its musical structure between the root songs and its versions. Music summaries are generated based on the following two criteria,

- (1) The selected segments are repeated at least once in the whole song.
- (2) The selected repeated groups should hold the majority of the song duration compared with other repeated groups.

Since all the repeated segments within the same group have approximately the same length, we calculate the total length that each label subsumes in a piece by multiplying the length of its one segment with its total number of segments, n . With the above mentioned selection criteria, we select one segment from each of the first two groups, which holds the longest duration of the song, to compute music summaries. Finally, we extract a fixed duration, l seconds, from each selected segment based on their starting-time information.

Research work by Gómez [Gómez06b] provides a successful example of version identification by means of analyzing the similarity of tonal features between music pieces. Thus, following previous research work [Gómez06b], we compute the instantaneous evolution of HPCP for both short summaries extracted from each song query and all the songs in the database. In order to measure similarity between two pieces, we apply the Dynamic Time Warping (DTW) algorithm [Ellis05], which estimates the minimum cost required to align one piece to the other one, on short summaries belonging to both pieces alternately as shown in Figure 6.1. DTW, also called Dynamic Programming (DP), is a widely-used method for performing dynamic time alignment and similarity measurements between two sequences that may vary in time and speed. Here, we can see that there appear four similar measures for each pair of comparisons. Finally, we choose the highest similarity among the four values to represent the similarity estimation between two pieces.

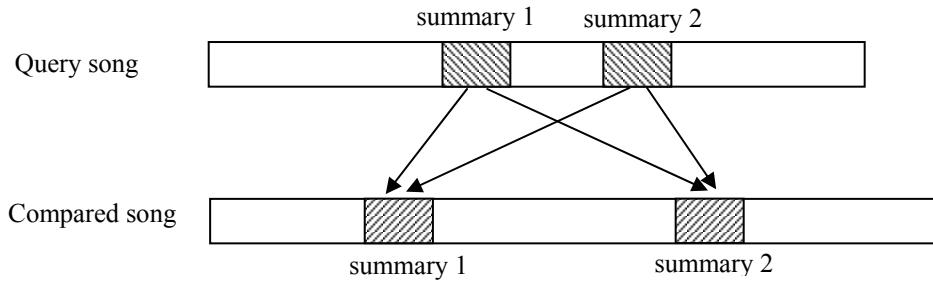


Figure 6.1. The comparison of summaries between two songs.

Since DTW actually performs a direct comparison between summaries from both pieces and considers that versions of the same piece do not necessarily maintain the same key (key change) as the original, we need to transpose the compared summaries to the same key as the query before computing similarity. One of the advantages of the octave equivalence tonal descriptors is that ring shifting of the feature vectors, which will be named $v_{compared}$, correspond to the transposition in music perception. Since a higher resolution of HPCP, with each coefficient corresponding to one third of a semitone is used, ring shifting three coefficients of the features vectors resembles transposing one semitone downwards for keys with same mode, or four semitones downwards for major-to-minor modes, or two semitones upwards for minor-to-major modes as illustrated in the circle of fifths' geometrical space (see Figure 6.2).

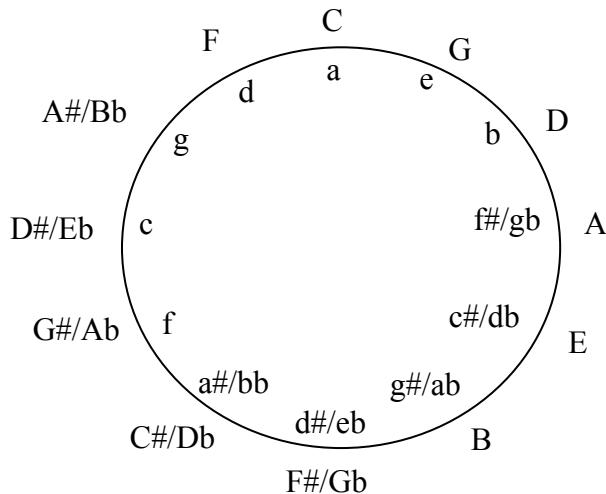


Figure 6.2. The circle of fifths geometry with major and minor modes. The major key for each key signature is shown as a capital letter on the outer circle whereas the minor key is shown as a small letter surrounded by the inner circle.

Thus, we can easily accomplish the task of transposing the summaries to the desired key by

$$v_{compared_modulated} = \begin{bmatrix} v_{compared}(:, Index_{shift} + 1:36) & v_{compared}(:, 1:Index_{shift}) \end{bmatrix} \quad (6.2)$$

$$\text{where, } Index_{shift} = 3 * (r - 1) \quad (6.3)$$

and $r \in \{1, 2, 3, \dots, 12\}$ denotes the $(r-1)$ number of semitones to be modulated downwards. We generated 12 different sets of the shifted feature vectors for each compared summary to evaluate the similarity between the query summaries and the 12 semitone transpositions of the compared summaries. It is noted that when $r=1$, no modulation occurs in the compared summary. Following that, we apply the DTW algorithm to query summaries and each 12 transposed compared summaries alternately to estimate the minimum cost between two summaries. Finally, the lowest estimated minimum cost is selected to represent the similarity between two songs. Figure 6.3 below illustrates the estimated minimum cost of the song summaries between a root query (song entitled *Imagine*) corresponding to 12 possible transpositions of its versions. As shown in the bar charts, versions sung by Diana Ross (in the key of F major) and Khaled & Noa (in the key of Eb major) achieve the lowest minimum cost at five semitone and three semitone downward transpositions respectively among the 12 possible transpositions, whereas the instrumental version, which has the same key as the root query, obtains the minimum cost at 0 transposed semitones.

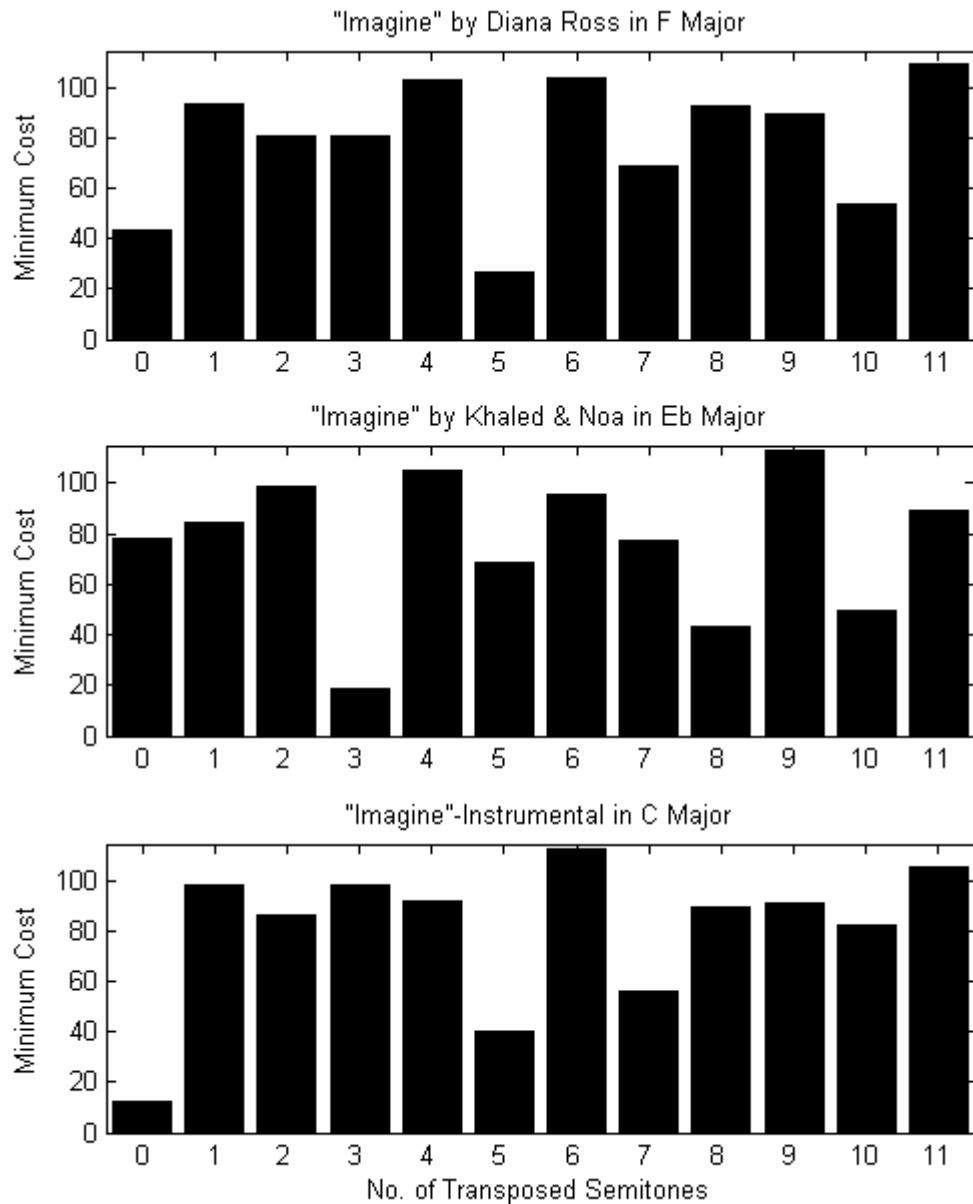


Figure 6.3. The estimated minimum cost of the song summaries between a root query (song entitled *Imagine*) corresponding to 12 possible transpositions of its versions. (1) *Imagine* sung by Diana Ross in F Major (2) *Imagine* sung by Kholed and Noa in Eb Major (3) Instrumental version of *Imagine* in C major.

6.2. Evaluation

In the following sections, we will describe in detail regarding our evaluation procedure. In addition, we discuss pros and cons of using our approach in song version identification.

6.2.1. Dataset

The goal of this study is to evaluate the applicability of structural descriptions in identifying different versions of a piece of music. Thus, we reuse the dataset described in [Gómez06b], which consists of 90 versions from 30 different songs (root query) of popular music as our test set. For this evaluation, we will compute a similarity measure between two different pieces based on low-level tonal descriptors, i.e. HPCP values. We will compare the efficiency of version identification obtained through the full length of the song with the one obtained through the song summaries.

6.2.2. Quantitative Measurements

Version identification, which involves song query and retrieval, is a type of information retrieval system. Thus, for evaluation purposes, we use IR standard measures, such as recall and precision, to rate effectiveness of the retrieval. The recall rate is defined as the ratio of the number of relevant returned documents to the total number of relevant documents for the user query in the collection. That is

$$\text{Recall rate} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|} \quad (6.4)$$

The precision rate is the ratio of the number of relevant returned documents to the total number of documents for a given user query. That is,

$$\text{Precision rate} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|} \quad (6.5)$$

To investigate the influence of the length, l , of the short summaries on the performance of version identification, we extract various durations from the range of 15 seconds to 25 seconds with an interval of 5 seconds from the audio signal. To estimate the optimal or upper bound performance of using summaries in version identification with our test set, we manually select two short segments (approximately 25 seconds depending on the tempo of the music), which are repeated in all the versions of the same songs, according to their time-varying harmonic contour in the segments. We substitute the manually selected segments for short summaries extracted based on music structural

descriptions to represent the song itself. Whereas for estimating the lower bound of the performances with the use of the short-summary approach, we randomly select two 25-seconds short segments for each song in the test set to represent the music itself. Finally, we compute similarity measures using the randomly selected or manually selected short segments. As explained above, we then select the highest similarity among the four values to represent the similarity estimation of the root query and the compared song.

6.2.3. Results

Here, we use precision-recall curves to capture the performance ranking of the version identification process. Figure 6.4 shows the performance of version identification using various numbers of short summaries extracted from the songs in different segments' lengths. From our results, we observe that the best performance is in the case of 25-seconds with two segments, which achieves a high precision and recall rates of 55.1% and 32.8%, respectively. As expected, the performances become impaired when the extracted summaries from the audio signals are decreased in length. For the case of 20-seconds, the performance achieves the precision and recall rates of 46.7% and 27.6%, respectively, whereas for the case of 15-seconds, the performance only scores 43.3% and 24.4% in its precision and recall measures. For the case where repetitiveness emphasis is applied on the short-summary approach, where only one 25-second summary is extracted from the songs, the achieved precision rate is the lowest, 36.7% with a recall level of 18.1%.

Figure 6.5 shows the performance of version identification using the whole-song approach versus the short-summary approach. From the precision-recall graph, we observe that by using two extracted short summaries (with the length of 25 seconds each) from the songs, we can achieve a slightly better performance in version identification compared with using the whole length of the piece. By only considering the first retrieved song for a given user query, the short-summary approach exceeds 0.6% and 2% in its precision and recall rates respectively compared with the whole-song approach. The estimated upper bound results for identifying different versions of the same song reaches the precision and recall rates of 66.6% and 36.8%, respectively. Whereas by using randomly extracted short summaries from songs, the achieved precision rate is very low, 22.2% with a recall level of 9.0%. The short-summary approach, besides its better accuracy compared with the whole-song approach, also consumes less time in performing version identification tasks. For our test set, which consisted of 90 audio data with an average audio length of 3 minutes and 45 seconds, the short-summary approach accomplishes the identification task at least 33% faster than the other approach. Figure 6.6 plots the average F-measures obtained from both approaches considering various numbers of songs for a given query. The statistical t-test shows that the obtained average F-measures from the

short-summary approach is significantly higher than those from the whole-song approach with the test result of $t(19)=3.966$, $p<0.01$ beyond the 99% confidence level.

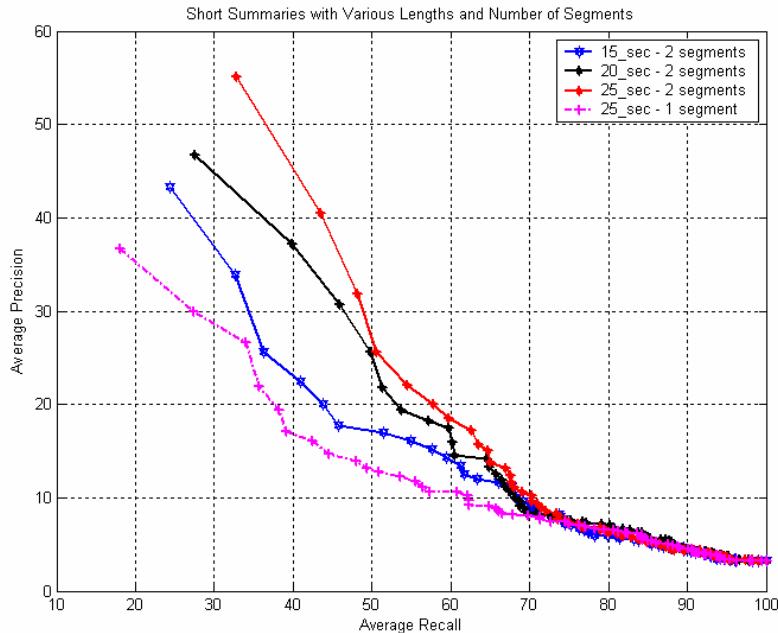


Figure 6.4. The performances of version identification by using various numbers of short summaries of different lengths based on its average precision and recall measures.

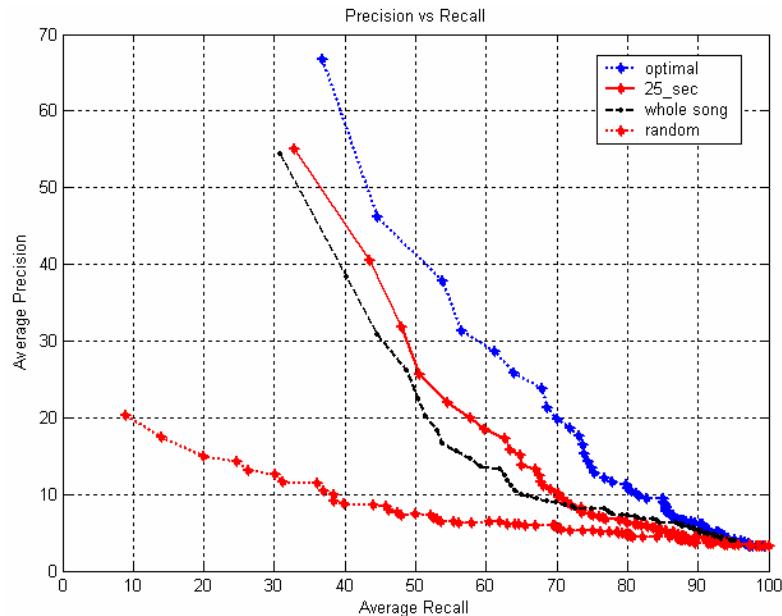


Figure 6.5. The performances of version identification: whole-song approach vs. short-summary approach based on its average precision and recall measures.

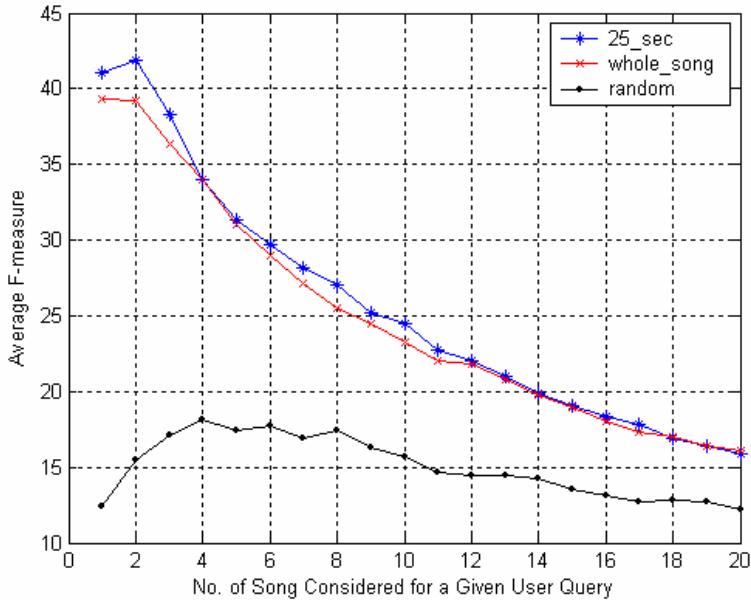


Figure 6.6. Average F-measures of both approaches (short-summary and whole-song) in version identification according to the number of songs considered for a given user query.

The bar graph in Figure 6.7 shows the average F-measures obtained for each retrieved song in the database, by considering the first 10-retrieved songs for a given user query. The song entitled *Imagine* achieves the overall best song performance with respect to identifying its different versions. Five song versions of *Imagine* are included in the test database: John Lennon (SongID-1), Tania Maria (SongID-6), Khaled and Noz (SongID-24), Diana Ross (SongID-40) and an instrumental version of *Imagine* (SongID-46). As reported in [Gómez06b], there are various musical differences between the *Imagine* song versions and its root song (SongID-1). These differences include noise, instrumentation, tempo, transposition, harmonization and structure. By considering the first 3-retrieval songs, querying using SongID-1, SongID-24, SongID-40 and SongID-46 achieves recall scores as high as 75%. In other words, 3 out of 4 song versions of *Imagine* appear in the first three considered retrieval results. The lowest retrieval performance of “*Imagine*” appears to be SongID-6 by Tania Maria. This version of “*Imagine*” is first performed in a fairly straightforward manner (with very melodic piano breaks). It is then broken into a quicker, celebratory samba groove. In addition, its music harmonization is very different from the rest of the “*Imagine*” versions. Thus, our algorithm finds it difficult to retrieve this version of *Imagine*.

Through observing the retrieval performance of each song, we notice that the song retrieval process achieves a better performance when querying different versions of the same song using the root (original) song instead of cover songs. Logically, this is reasonable, since song versions tend to

imitate or mimic the musical properties of the original version. Thus by querying the root (original) song, there is a higher possibility of comparing specific segments from the song version that mimicked the original version, thus increasing the accuracy of the song version identification task.

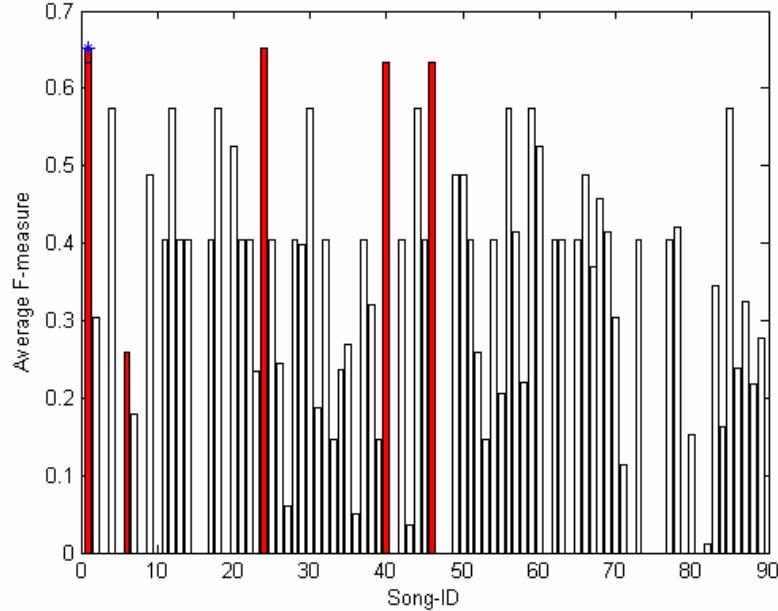


Figure 6.7. Average F-measures obtained for each retrieved song in the database with the considered first 10-retrieved songs for a given user query. Descriptions: Filled bars mark the cover songs of *Imagine* by different artists, whereas SongID-1 marked “*” denotes the root song, *Imagine* by The Beatles.

Through analyzing the low performance of a few query songs, we have realized that there occurs an issue with regards to the transitivity relationship between songs due to our two extracted short summaries comparison approach. The following section uses the given example as illustrated in Figure 6.8 to give a better explanation regarding this issue of the transitivity relationship between songs. For instance, if Song-A has two summaries with each appearing in Song-B and Song-C, by querying Song-A, we will be able to find both Song-B and Song-C as its versions. However if Song-C happens to have summaries which appear one in Song-A but none in Song-B, by querying Song-C, we will only find Song-A but miss Song-B since we do not infer any relationship between songs. Nevertheless, the failure in this aspect could be exploited or considered interesting for generating an additional source of metadata that is not directly stored in the database. Seeing that cover songs tend to imitate the original song, by inferring the transitivity relationships among different versions of the same song, it would provide clues to defining the original song among its different versions. For instance, in the above given example, the present of version relationships of Song-B and Song-C with

Song-A respectively but not within themselves (Song-B and Song-C) may imply that Song-A could be considered the canonical song (the original song version).

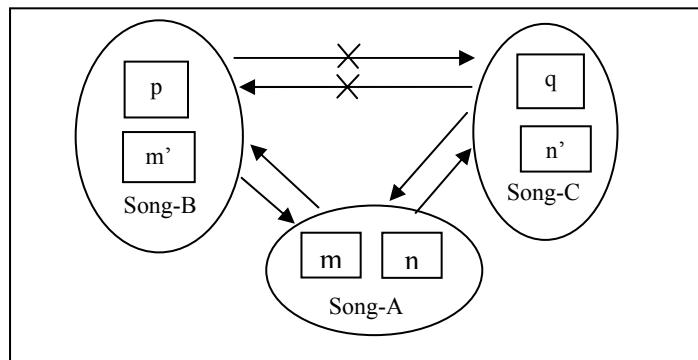


Figure 6.8. Transitivity relationship between songs.

6.2.4. Discussion

In the experiment results, perhaps the most notable result from this chapter's experiment is the distinctive dichotomy in performance between the two distinct selection criteria (repetitiveness emphasis vs. repetitive equivalence emphasis) in identifying representative excerpts of music for version identification applications. For the segment selections that make use of a complementary musical knowledge (i.e. repetitive equivalence emphasis), we see generally good performance. This dichotomy generally supports the notion that repetitiveness of music segments is important in identifying representative excerpts of music. However it is not the only assumption that we should rely on, depending on the application context. Incorporating musical knowledge related to the processing data (e.g. structural differences within the song versions) somehow improves performance.

Finally, as a conclusion of this small-scale of evaluation, we can also see that the short-summary approach seems to perform better than the whole-song approach in both retrieval accuracy and computational efficiency. From this study, we have observed a few advantages and disadvantages of using the short-summary approach in version identification compared with the whole-song approach. The advantages are:

- (i) Time consumption factor – less time consuming and higher identification performance for the database, which consists of songs with long durations;
- (ii) Modulation within piece – since only two short segments are extracted from the song itself, the performance accuracy is not to be affected by modulation within the pieces;

- (iii) Different music structural descriptions in song versions – flexible to structural changes since only the core segments are extracted from the music itself;

Whereas, the disadvantages of using such an approach include:

- (i) Identifying a song and its versions with large tempo variances – since short and fixed time constraints are applied in extracting summaries from the song, false negatives may occur for the query and its versions which have large differences in tempo;
- (ii) Songs with short duration – applying such an approach to songs with durations shorter than double the extracted summaries length is more time consuming than the whole-song approach;

6.3. Summary

In this chapter, we have investigated the potential of using music structural descriptions in identifying different versions of a piece of music. We have presented our approach in extracting significant excerpts from music audio files for version identification applications. Experiments were conducted to evaluate and compare the performance of our proposed approach with other approaches from previous work [Gómez06b]. From the experiment results, we have shown evidence of the utility of our approach in the form of retrieving different versions of a song in a music collection but still leaves room for additional enhancement.

In the next and final chapter, we will present a summary of the main conclusions from this work. Additionally, we will present suggestions for improvement, open questions and potential areas for our future work presented in our PhD dissertation.

Chapter 7

Conclusions and Future Work

Within this dissertation we have addressed four main aspects related to describing structural content from music signals: the detection of acceptable structural boundaries related to section changes in music, the analysis and discovery music structures via inferring repeated patterns that appear in the music, the identification of significant audio excerpts to represent a piece of music together with human-based subjective evaluation, and finally how music structural description facilitates in the identification of song versions in music collections.

The goal of this chapter is to summarize the contributions this dissertation makes to the current state of the art in structural analysis and segmentation of music signals. This is followed by the main conclusions that we have drawn from our research. Finally, we present some suggestions and ideas for future work in this field.

7.1. Summary of Contributions

In this research work, we have fulfilled our initial goal of studying and developing algorithm frameworks and methods in two areas that are closely related to automatic audio-based music structural analysis: (i) Semantic audio segmentation; and (ii) music structure discovery and high-level music description. With the extracted music structural description, we have also completed our goal in identifying “singular” within-song excerpts in popular music by proposing a new method in defining representative excerpts from music signals. With regards to the applicability of structural descriptions in the music information retrieval context, we have carried out the task of identifying different song versions of the same songs by introducing a novel retrieval concept which is based on high-level

descriptions of music. We also include our significant contributions with regards to current state of the art in structural analysis and segmentation of music signals.

In the literature review chapter, we have reviewed current literature related to structural analysis and segmentation by studying the similarity and differences between these approaches and discussing their advantages and disadvantages in performing the related tasks.

In our semantic audio segmentation study, we have proposed a two-phase approach to segment audio data according to the structural changes of music and to provide a way to separate the different music “sections” of a piece, such as the “intro”, “verse”, “chorus”, etc. We have also proposed a combination set of audio descriptors that has proved useful in detecting music structural changes. In addition, we have also utilized higher-level analysis techniques, such as beat detection, to improve the accuracy of the structural boundary detection process. Evaluation tests have been carried out to assess the performance of our proposed method with the use of a test dataset consisting of 54 pop songs. From the quantitative evaluation results, we conclude that the exploitation of image processing techniques (i.e. morphological filtering) is significantly profitable in enhancing the detection of segment boundaries corresponding to the structural changes and in facilitating semantic segmentation of music audio. By having two phases of segmentation, first focusing on rough segmentation and later on further refinement, we have yielded a semantic audio segmentation algorithm that is useful and relatively reliable for practical applications. Coupling semantic audio segmentation functionality into both hardware and software platforms of digital audio players or sound visualization and manipulation applications will allow users to skip from one section to another section of music easily and precisely. This is certainly a big improvement over the conventional fast-forward function mode.

In our music structural analysis and discovery study, we have further improved upon the existing method for detecting chorus sections in music [Goto03a] to produce a complete and unified high-level structural description directly from music signals. Instead of only discovering the most repeated sections that appear in music, we have also identified (dis)similar sections in music by tagging these with explicit labelling. Herein, we have investigated and compared the applicability of different tonal-related features used in music structural discovery. We have also proposed the use of timbre-based semantic audio segmentation to rectify the common boundaries inaccuracies appearing in music structural descriptions obtained by means of depending on single tonal-related features to discover musical structure from acoustics signals. In order to obtain proper structural descriptions of music, we have addressed the problem of tackling the complexity of modulation within a song that has not much been addressed by the existing music structural discovery algorithms. Evaluation tests were done on three different databases consisting of more than 100 pop songs in various languages from different

regions in the world, to assess the applicability of our approach to real world popular music. Our proposed approach achieved overall precision and recall rates of 79% and 85%, respectively, for correctly detecting significant structural boundaries in the music signals from the above described three datasets. Compared to an existing structural analysis system [Chai03c] on their same test data, our proposed approach obtains slightly better performance in its quantitative evaluation results. From the quantitative evaluation results, we conclude that our integration of timbre-based semantic audio segmentation to our music structural description system significantly improves its effectiveness in detecting structural descriptions of music. Coupling music structural descriptions into audio playback devices will allow fast browsing of music data. In addition, an auxiliary add-in playback mode, in combination with the segment block structural visualization, will allow users to have easy access to any particular segment of the music by just clicking on the visualization's segment block. Figure 7.1 shows an example of a sound visualization system coupled with music structure visualization and add-in segment playback functionalities.

For the identification of the representative excerpts of music, we have considered other factors in the identification task than the ones appearing in the literature. Our hypothesis states that repetitiveness of music may not be the only element in detecting the retrieval cue of music. Thus, we proposed our novel segment-to-song approach, in which the high similarity between repetitive segments and the entire song is considered, to identify the representative excerpt of the music. An online listening test has been conducted to obtain some subjective evaluation of our approach, based on human perception. A database of 18 music tracks comprising popular songs from various artists was used in the subjective evaluation. From the objective evaluation results, we conclude that our proposed segment-to-song approach captures the most song titles with its extracted representative excerpts compared to the other two investigated approaches (i.e. most-repetitive approach and first-30-seconds approach). The subjective evaluation results show that participants are able to correctly identify the song titles much more easily with the presented extracted excerpts based on our segment-to-song approach. For evaluating subjects' preference for specific approaches in extracting good music summaries, the obtained subjective evaluation results indicate a strong dependency on the subjects' musical backgrounds. Specifically, subjects with stronger musical backgrounds prefer our proposed approach (segment-to-song) over the most-repetitive approach or the first-30-seconds approach in extracting a song summary. In contrast, for the case of subjects with none or basic musical backgrounds, they prefer the most-repetitive approach over the other two approaches. This result encourages the consideration of other factors, which have not yet been explored by current references, to further investigate the identification of representative excerpts of music. These factors can be such as the distinctive sound of a strummed guitar or a drum roll that appears in the beginning of a music piece and so forth. On the other hand, since our approach is based on music structural

descriptions, we are able to exploit the obtained information to visualize the structure of music. Thus, the identification of representative music excerpts does not only bring the benefit of giving an abstraction cue, but also a clear visual-structural representation of the music.

For music structural description in song versions identification, we have introduced a new concept for music retrieval. Our hypothesis stated that prior knowledge of the structural descriptions of the music would give a better grasp of the musical content and contribute to efficient retrieval of large amounts of digital audio data. Thus, instead of using the entire music piece to find how well two compared pieces match, we have proposed using the extracted short excerpts from the music signals based on our prior knowledge of their music structural descriptions to find different versions of the same song. A song database, consisting of 90 versions from 30 different songs of popular music, was used to evaluate the performance of our proposed approach. Quantitative results have confirmed the validity of our proposed concept by showing an explicit improvement to accuracy and time-saving factors for the song version identification task compared to previous research work [Gómez06b] using the same test set. This result encourages the consideration of using structural-based extracted audio excerpts instead of entire songs in some other areas related to music content analysis and processing, such as content-based similarity for music recommendation purposes.

7.2. Conclusion

Our general conclusion obtained by reviewing the current state of the art in this area states that there exist a few limitations with respect to algorithm evaluation in the current literature. The first limitation is the lack of generality of the test databases and solid ground truths for algorithm evaluation. Great human resources and efforts are required just to obtain just a small set of ground truth for algorithm evaluation. Goto's shared database (DB) compiled specifically for research purposes, *RWC Music Database* [Goto03c], is a good attempt at enabling researchers to compare and evaluate their various systems and methods against a common standard. Unfortunately, the preparation of this shared database does not consider music structural analysis related issues. Thus, there is no solid ground truth that can be acquired directly from the shared database. MTG's attempt at a Music Content Semantic Annotator (MUCOSA) [Herrera05], an annotation environment that allows users to semi-automatically annotate music content description from low-level to semantic labels, could be a way to obtain a solid ground truth algorithm evaluation in this specific area, even though issues regarding how to avoid any violations of the use of audio resources from the client would need to be seriously considered.

Another limitation is the method used to weight the importance of extracted music sections. Much of research studies in this area either provides no evaluation regarding the significance of the

extracted music sections or only depends on the presence of chorus/refrain sections to define the quality of the music excerpts. The significance of the musical excerpts in audio signals highly depends on human perceptions. Thus, subjective evaluation based on the human perception should be taken into consideration to evaluate the significance of the extracted music sections. Factors such as listeners being musicians or non-musicians who may not have the same viewpoint on “which sections are the representative excerpts of a piece of music” also need to be considered. For instance, a musician may have a strong impression of the solo instrumental sections whereas this may not be the case for a non-musician. Hence, it would be useful to have two groups of listening subjects and to take into consideration the differences between these two groups when evaluating the significance of the extracted music sections as well.

7.3. Future Work

In this research work, the generality of our music database is quite limited. So far, we have limited our scope to only “pop” music and have not tested our approach on different music genres, such as instrumental music, jazz, or classical music. Thus, in our future work, we will take into consideration some other different music genres that we have not yet explored, in order to assess our proposed method on a wider generality of music applications.

With regards the semantic audio segmentation aspect, it is worthwhile to pay attention to the fact that the precision and recall rates of our proposed segmentation method are particularly low for those songs that include smooth transitions between sections. It seems that our descriptors are not sensitive enough to mark these changes. On the other hand, songs with abrupt transitions between sections usually achieve better rates on these measures. Thus, using some other disregarded descriptors, perhaps we will be able to cope with this matter.

In music structure discovery and high-level description aspects, an interesting direction for future work would be to use computed music structural descriptions to automatically label sections according to their structural titles, such as intro, verse, bridge, outro, etc. By doing this, we can provide a more informative description of the structure of music. In addition, by visualizing music structural descriptions in audio editing applications coupled with click-and-play mode functionality, the structure of music will also be visualized on the screen display and allow users to have easy access to any particular section by just clicking on the displayed section-block as shown in Figure 7.1. This will definitely facilitate better audio browsing and retrieval. The first step towards this goal would be to comprehend the typical structure of the kind of music we are interested in dealing with and to construct well-formed rules to define the structural titles.

It is clear that identifying representative musical excerpts of audio files has relevance to music summarization. In this research work, we only identify a significant representative excerpt from a piece of music. Thus, our future plans include making use of our structural analysis algorithm to generate summaries of music, in which different structural sections that appear in the music piece will be combined, to give an overview summary representation of the music pieces as those proposed by [Peeters02]. Subjective evaluation will then be used to study the quality of differences in music summaries, generated using the combination of various sections versus a single audio excerpt, based on human judgment.

For music structural descriptions in the song version identification aspect, we have only explored one among other approaches in finding different versions of the same songs. In future, it would be interesting to explore other song versions identification techniques but utilizing the same concept in which the short excerpts instead of the entire song are used to make comparison between songs.

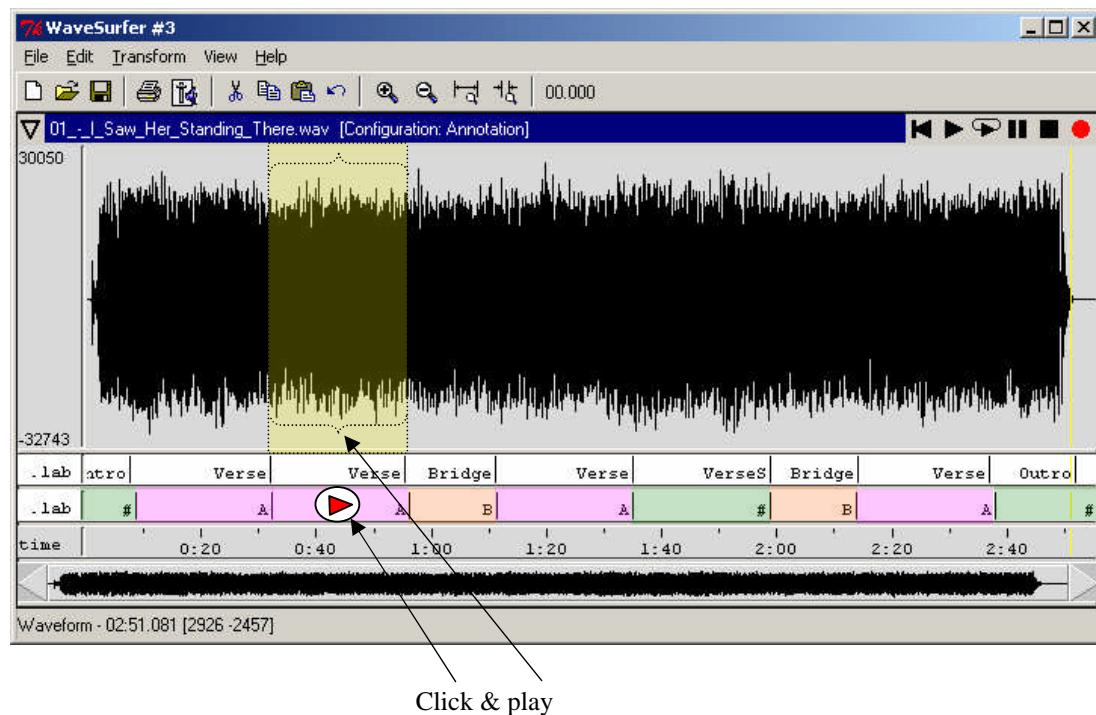


Figure 7.1 An example of a sound visualization system coupled with music structure visualization and add-in segment playback functionalities.

Finally, another future direction includes incorporating our algorithm into practical applications. Since our music structural analysis approach begins with the identification of repetitions that appear in the music, we are able to exploit the obtained information to visualize the structure of music.

Presumably, coupling music structural description information into song similarity applications will not only allow finding of similar songs but also finding of similar song segments. In addition, structural similarity between songs could be an additional factor to be considered for finding similar songs. Figure 7.2 shows an example of a finding song similarity system coupled with music structural visualization, including segment search by similarity and playback functionalities.

7.4. Final Thoughts

Traditionally, music signals were represented as a mixture of sinusoidal signals by digital audio editing applications such as, Adobe Audition⁸ and SoundForge⁹. By segmenting and discovering the structural patterns appearing in music, we are providing additional plus novel information on the music signals that bridge the gap between low-level and higher-level of music descriptions. Practically, these new descriptions of music promote new ways of dealing with music signals. A straightforward example would be in areas related to music signals visualization. For instance, digital sound editing applications would no longer be limited to only visualizing music signals as chaotic waveforms but as a series of symbols or colourful block representations that somehow show the subsumed structural content of the music signals. Without doubt, such information would be more useful and much easier for users to comprehend. In addition, the ability to gain direct access to the structural level of music instead of merely plain waveforms would be very useful for practical applications in music information retrieval contexts, such as audio indexing, audio browsing, and audio database management. The presented overview is expected to have made helpful contributions to the development of such applications.

⁸ Adobe webpage: <http://www.adobe.com/special/products/audition/syntrillium.html>

⁹ Sony Media Software webpage:

<http://www.sonymediasoftware.com/Products>ShowProduct.asp?PID=961>

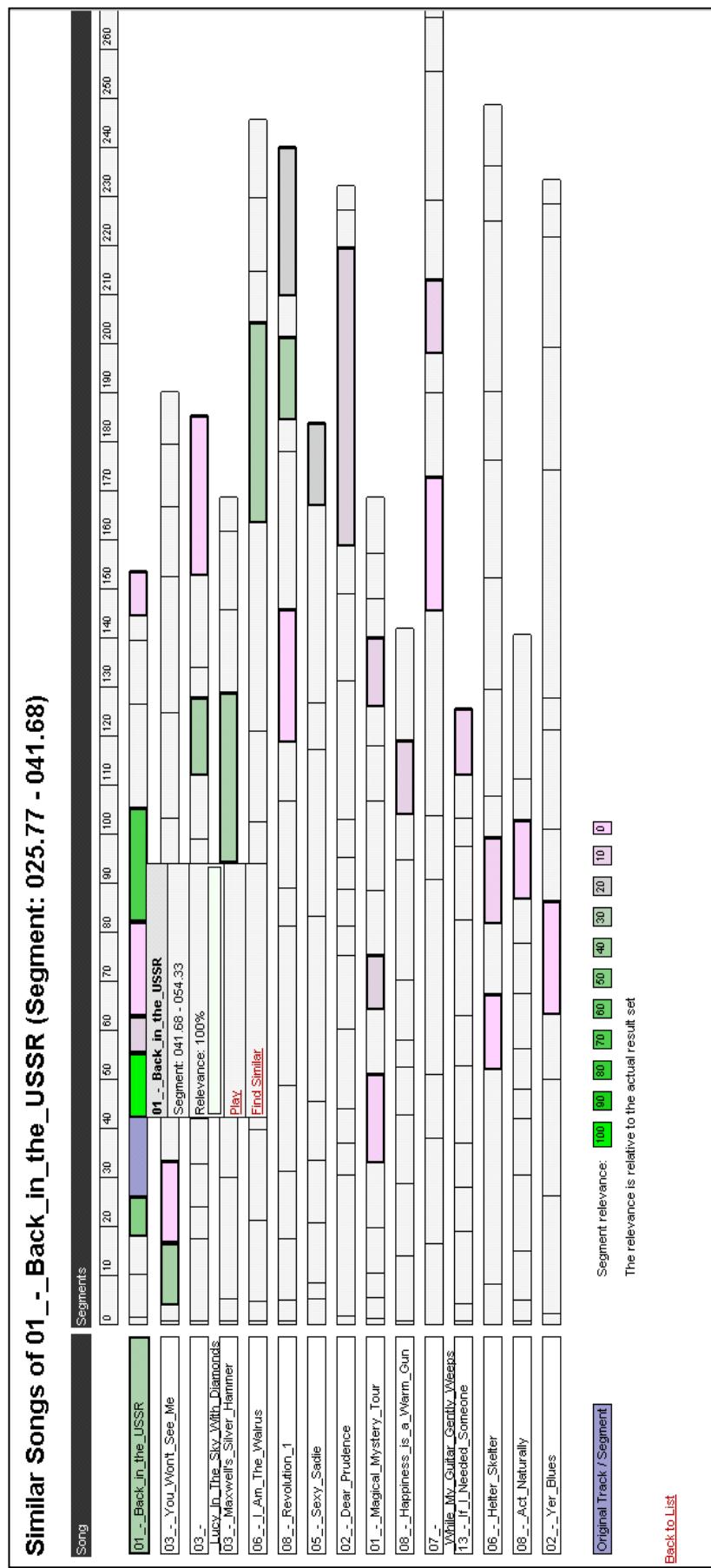


Figure 7.2. An example of finding song similarity system coupled with music structural visualization, add-in finding segment similarity and playback functionalities

Bibliography

- [Abdallah05] Abdallah, S., Casey, M., Noland, K., Sandler, M., and Rhodes, C. Theory and Evaluation of a Bayesian Music Structure Extractor. *Proc. International Conference on Music Information Retrieval (ISMIR)*, London, 2005.
- [Adam03] Adams, W. H., Lyengar, G., Lin, C-Y, Naphade, M. R., Neti, C., Nock, H. J., and Smith, J. R. Semantic Indexing and Multimedia Content Using Visual, Audio, and Text Cues. *EURASIP Journal on Applied Signal Processing*, vol. 2, pp. 170-185, 2003.
- [Allen90] Allen, P., and Dannenberg, R. Tracking Musical Beats in Time. In *Proceedings of the 1990 International Computer Music Conference*, pp. 140-143. San Francisco: International Computer Music Association, 1990.
- [Arons93] Arons, B. SpeechSkimmer: Interactively Skimming Recorded Speech. *ACM Symposium on User Interface Software and Technology (UIST'93)*, ACM Press, pp. 187-196, 1993.
- [Aucouturier01] Aucouturier, J.-J. and Sandler, M. Segmentation of Musical Signals Using Hidden Markov Models. *AES 110th Convention*, Amsterdam, the Netherlands, 2001.
- [Aucouturier02] Aucouturier, J.-J. and Sandler, M. Finding Repeating Patterns in Acoustic Musical Signals: Applications for Audio Thumbnailing. *AES 22nd International Conference on Virtual, Synthetic and Entertainment Audio, Espoo*, Finland, 2002.
- [Baker89a] Baker, M. An Artificial Intelligence approach to Musical Grouping Analysis. *Contemporary Music Review*, vol. 3, pp. 43-68, 1989.
- [Baker89b] Baker, M. A Computational Approach to Modeling Musical Grouping Structure. *Contemporary Music Review*, vol. 4, no. 1, pp. 311-325, 1989.
- [Bartsch05] Bartsch, M. and Wakefield, G. Audio Thumbnailing of Popular Music Using Croma-Based Representations. *IEEE Transactions on Multimedia*, vol. 7, no. 1, 2005.
- [Bartsch01] Bartsch, M. and Wakefield, G. To Catch A Chorus: Using Chroma-Based Representations for Audio Thumbnailing. *IEEE Workshop on Applications of*

- Signal Processing on Audio and Acoustics (WASPAA)*, New Paltz, New York, USA, 2001.
- [Bharucha87] Bharucha, J. J. Music Cognition and Perceptual Facilitation: A Connectionist Framework. *Music Perception*, vol. 5, pp. 1-30, 1987.
- [Bharucha91] Bharucha, J. J. Pitch, Harmony and Neural Nets: A Psychological Perspective. In P. M. Todd & D. G. Loy (Eds.), *Music and Connectionism*, pp. 84-99. Cambridge, MA: MIT Press, 1991.
- [Birmingham01] Birmingham, W. P., et al. MUSART: Music Retrieval via Aural Queries. *Proceedings Second International Symposium on Music Information Retrieval*, pp. 73-81, 2001.
- [Boltz86] Boltz, M. and Jones, M. R. Does Rule Recursion Make Melodies Easier to Reproduce? If not, what does? *Cognitive Psychology*, vol. 18, pp. 389-431, 1986.
- [Brown91] Brown, J. C. Calculation of a Constant Q Spectral Transform. *Journal of the Acoustical Society of America*, vol. 89, pp. 425-434, 1991.
- [Burgeth04] Burgeth B., Welk M., Federn C., and Weickert J. Morphological Operations on Matrix-Valued Images. *The 8th European Conference on Computer Vision*, pp. 155-167, Prague, Czech, May 2004.
- [Burns87] Burns, G. A Typology of ‘hook’ in Popular Records. *Popular Music*, vol. 6, no. 1, pp. 1-20, 1987.
- [Chai03a] Chai, W. and Vercoe, B. Music Thumbnailing via Structural Analysis. *Proceedings of ACM Multimedia Conference*, November 2003.
- [Chai03b] Chai, W. and Vercoe, B. Structural Analysis of Musical Signals for Indexing and Thumbnailing. *Proceedings of ACM/IEEE Joint Conference on Digital Libraries*, May 2003.
- [Chai03c] Chai, W. *Structural Analysis of Musical Signals for Indexing, Segmentation and Thumbnailing*. Paper for the Major Area of the PhD General Exam, March 2003.
- [Chai03d] Chai, W. Structural Analysis Of Musical Signals via Pattern Matching. *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, April 2003.
- [Chai05] Chai, W. *Automated Analysis of Musical Structure*. PhD Dissertation, MIT, 2005.
- [Cheng03] Cheng, Y. *Content-Based Musical Retrieval on Acoustical Data*. Ph.D. thesis, Stanford University, August 2003.
- [Cooper02] Cooper, M. and Foote, J. Automatic Music Summarization via Similarity Analysis. *International Symposium on Music Information Retrieval*, Paris, France, 2002.

- [Cooper03] Cooper, M. and Foote, J. Summarizing Popular Music via Structural Similarity Analysis, *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, October 19, 2003.
- [Croonen94] Croonen, W. L. Effects of Length, Tonal Structure, and Contour in the Recognition of Tone Series. *Perception & Psychophysics*, vol. 55, pp. 623–632, 1994.
- [Dannenberg02a] Dannenberg, R. B. and Hu, N. Discovering Musical Structure in Audio Recordings. *Proc. 2nd Int. Conference in Music & Artificial Intelligence (ICMAI)*, 2002.
- [Dannenberg02b] Dannenberg, R. B. Listening to 'Naima': An Automated Structural Analysis of Music from Recorded Audio. *Proceedings of the 2002 International Computer Music Conference*, pp. 28-34, 2002.
- [Dasgupta06] Dasgupta, S., Papadimitriou, C. H., and Vazirani, U. V. *Algorithms*. McGraw-Hill Higher Education, 2006.
- [Davies98] Davier, G. and Thomson, D., eds. *Memory in Context: Context in Memory*. Wiley: Chichester, England, 1998.
- [Davies05] Davies, M. E. P. and Plumley, M. D. Beat Tracking with a Two State Model. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2005.
- [Deutsch80] Deutsch, D. The Processing of Structured and Unstructured Tonal Sequence. *Perception & Psychophysics*, vol. 28, pp. 381-389, 1980.
- [Dowling73] Dowling, W. J. Rhythmic Groups and Subjective Chunks in Memory for Melodies. *Perception & Psychophysics*, vol. 14, pp. 37-40, 1973.
- [Dowling78] Dowling, W. J. Scale and Contour: Two Components of a Theory of Memory for Melodies. *Psychological Review*, vol. 85, pp. 342-354, 1978.
- [Duxburg02] Duxburg, C., Sandler, M., and Davies, M. A Hybrid Approach to Musical Note Onset Detection. *International Conference of on Digital Audio Effects (DAFx'02)*, 2002.
- [Edworthy85] Edworthy, J. Melodic Contour and Musical Structure. In P. Howell, I. Cross, & R. J. West (Ed.), *Musical Structure and cognition*, London: Academic Press Inc., pp. 169-188, 1985.
- [Ellis94] Ellis, G. M. *Electronic Filter Analysis and Synthesis*, Artech House, 1994.
- [Ellis05] Ellis, D. *Dynamic Time Warp (DTW) in Matlab*. Web publication, 2005. <http://www.ee.columbia.edu/dpwe/resources/matlab/dtw>
- [Filonov05] Filonov, A. S., Gavrilko, D. Y., and Yaminsky, I. V. *Scanning Probe Microscopy Image Processing Software User's Manual "FemtoScan"*. version 4.8. Moscow: Advanced Technologies Center, 2005.

<http://www.spm.genebee.msu.su/manual/en/node108.html>

- [Foote99] Foote, J. Visualizing Music and Audio Using Self-Similarity. *ACM Multimedia*, pp. 77–84, Orlando, Florida, USA, 1999.
- [Foote00] Foote, J. Automatic Audio Segmentation Using a Measure of Audio Novelty. *IEEE Int. Conf. Multimedia and Expo (ICME)*, vol. 1, pp. 452-255, New York City, NY, USA, 2000.
- [Foote03] Foote, J. and Cooper, M. Media Segmentation Using Self-Similarity Decomposition. *Proceedings of SPIE Storage and Retrieval for Multimedia Databases*, vol. 5021, pp. 167-175, 2003.
- [Friberg04] Friberg, A. A Fuzzy Analyzer of Emotional Expression in Music Performance and Body Motion. In J. Sundberg & B. Brunson (Eds.) *Proceedings of Music and Music Science*, Stockholm, 2004.
- [Fujishima99] Fujishima, T. Realtime Chord Recognition of Musical Sound: A System Using Common Lisp Music. *Proceedings of International Computer Music Conference (ICMA)*, pp. 464-467, 1999.
- [Gang02] Gang, Q., Sural, S., and Pramanik, S. A Comparative Analysis of Two Distance Measures in Color Image Databases. *IEEE International Conference of Image Processing*, vol. 1, pp. 22-25, 2002.
- [Gjerdingen94] Gjerdingen, R. O. Apparent Motion in Music? *Music Perception*, vol. 11, pp. 335-370, 1994.
- [Gómez03] Gómez, E., Klapuri, A., and Meudic, B. Melody Descriptions and Extraction in the Context of Music Content Processing. *Journal of New Music Research*, vol. 32, no. 1, 2003.
- [Gómez06a] Gómez, E. Tonal description of polyphonic audio for music content processing. *INFORMS Journal on Computing, Special Cluster on Computation in Music*, vol. 18, no. 3, 2006.
- [Gómez06b] Gómez, E. *Tonal Description of Music Audio Signals*. PhD thesis. UPF, Barcelona, 2006.
- [Goto99] Goto, M. A Real-Time Music Scene Description System: Detecting Melody and Bass Lines in Audio Signals. *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI) Workshop on Computational Auditory Scene Analysis*, pp. 31-40, 1999.
- [Goto00] Goto, M. Robust Predominant-F0 Estimation Method for Real-time Detection of Melody and Bass Lines in CD Recordings. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. II-757-760, 2000.
- [Goto03a] Goto, M. A Chorus-Section Detecting Method for Musical Audio Signals. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp.V-437-440, April 2003.

- [Goto03b] Goto, M. A SmartMusicKIOSK: Music Listening Station with Chorus-Search Function. *Proceedings of the 16th Annual ACM symposium on User Interface Software and Technology (UIST'03)*, vol 5, no. 2, pp. 31-40, 2003.
- [Goto03c] Masataka, G., Hiroki, H., Takuichi, N., and Ryuichi, O. RWC Music Database: Music Genre Database and Musical Instrument Sound Database. *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR 2003)*, pp. 229-230, October, 2003.
- [Gouyon03] Gouyon, F. and Herrera, P. A Beat Induction Method For Musical Audio Signals. *Proceedings of 4th WIAMIS-Special session on Audio Segmentation and Digital Music*, London, UK, 2003.
- [Herrera05] Herrera, P., Celma, O., Massaguer, J., Cano, P., Gómez, E., Gouyon, F., Koppenberger, M., Garcia, D. G., Maheder, J., and Wack, N. Mucosa: a music content semantic annotator'. *Proceedings of 6th International Conference on Music Information Retrieval*, London, UK, 2005.
- [Hofmann97] Hofmann, T. and Buhmann, J. M. Pairwise Data Clustering by Deterministic Annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 1, 1997.
- [Holtzmann77] Holtzmann, S. R. A Program for Key Determination. *Interface*, vol. 6, pp. 29-56, 1977.
- [Huron89] Huron, D. Voice Segregation in Selected Polyphonic Keyboard Works by Johann Sebastian Bach. Ph.D. diss., Nottingham University. 1989.
- [Huron99] Huron, D. *Humdrum User's Guide*, <http://dactyl.som.ohio-state.edu/Humdrum/guide.toc.html>, last updated 1999.
- [Huttenlocher97] Huttenlocher, J. and Prohaska, V. *Reconstructing the Times of Past Events. Memory for Everyday and Emotional Events*. Mahwah, NJ: Lawrence Erlbaum Associates, pp. 165-179, 1997.
- [Jain99] Jain, A. K., Murty, M. N., and Flynn P. J. Data Clustering: A Review. *ACM Computing Surveys*, vol. 31, no. 3, pp. 264-323, 1999.
- [Kim06] Kim, S., Kim S., Kwon, S., and Kim H. A Music Summarization Scheme using Tempo Tracking and Two Stage. *International Workshop on Multimedia Signal Processing (MMSP06)*. British Columbia, Canada, 2006. (poster).
- [Klapuri03] Klapuri, A. Automatic Transcription of Music. In *Proceedings Stockholm Music Acoustics Conference (SMAC 03)*, Stockholm, Sweden, 2003.
- [Krumhansl90] Krumhansl, C. L. *Cognition Foundations of Musical Pitch*. New York: Oxford University Press. 1990.
- [Large94] Large, E. W., and Kolen, J. F. Resonance and the Perception of Musical Meter. *Connection Science*, vol. 6, pp. 177-208, 1994.
- [Lee91] Lee, C. The Perception of Metrical Structure: Experimental evidence and a model. In P. Howell, R. West, and I. Cross (Eds), *Representing Musical*

- Structure*, pp. 59-127. London: Academic Press, 1991.
- [Leman95] Leman, M. *Music and Schema Theory*. Berlin: Springer, 1995.
- [Lerdahl83] Lerdahl, F., and Jackendoff, R. *A Generative Theory of Tonal Music*. Cambridge, MA: MIT Press, 1983.
- [Levy06a] Levy, M., Sandler, M., and Casey, M. Extraction of High-Level Musical Structure from Audio Data and its Application to Thumbnail Generation. *IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France, May 2006.
- [Levy06b] Levy, M. and Sandler, M. New Methods in Structural Segmentation of Musical Audio. *14th European Signal Processing Conference (EUSIPCO)*, Florence, Italy. 2006.
- [Liu04] Liu, H. Content-Based TV Sports Video Retrieval Based on Audio-Visual Features and Text Information. *Proceedings of the International Conference on Web Intelligence (WI'04)*, 2004.
- [Logan00] Logan, B. and Chu, S. Music Summarization Using Key Phrases. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, Istanbul, Turkey, 2000.
- [Logan01] Logan, B. and Salomon, A. A Music Similarity Function Based On Signal Analysis. *International Conference on Multimedia and Expo*, Tokyo, Japan, 2001.
- [Longuet-Higgins71] Longuet-Higgins, H.C., and Steedman, M. J. On Interpreting Bach. *Machine Intelligence*, vol. 6, pp. 221-241, 1971.
- [Lu03] Lu, L. and Zhang, H.-J. Automated Extraction of Music Snippets. *Proceeding of ACM Multimedia*, pp.140-147, 2003.
- [Lu04] Lu, L., Wang, M., and Zhang, H.-J. Repeating Pattern Discovery and Structure Analysis from Acoustic Music Data. In *6th ACM SIGMM International Workshop on Multimedia Information Retrieval (MIR' 04)*, pp. 275-282, 2004.
- [MacQueen67] MacQueen, J. Some Methods for Classification and Analysis of Multivariate Observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281-296, 1967.
- [Maddage04] Maddage, N. C., Xu, C., Kankanhalli, M. S., and Shao, X. Content-Based Music Structure Analysis with the Applications to Music Semantic Understanding. *ACM Multimedia Conference (ACM MM04)*, 2004.
- [Maddage06] Maddage, N. Automatic Structure Detection Popular Music. *IEEE Multimedia*, vol. 13, no. 1, pp. 65-77, 2006.
- [Marsden92] Marsden, A. Modelling the Perception of Musical Voices: A Case Study in Rule-Based Systems. In A. Marsden and A. Pope (Eds.), *Computer Representations and Models in Music*, 239-63. London: Academic Press. 1992.

- [Maxwell92] Maxwell, H. J. An Expert System for Harmonic Analysis of Tonal Music. In M. Balaban, K. Ebcioğlu, & O. Laske (Eds.), *Understanding Music with AI*, pp. 335–353. Cambridge, MA: MIT Press, 1992.
- [McCabe97] McCabe, S. L. and Denham, M. J. A Model of Auditory Streaming. *Journal of the Acoustical Society of America*, vol. 101, pp. 1611–1621, 1997.
- [Nam97] Nam, J., Cetin, A. E., and Tewfik, A. H. Speaker Identification and Video Analysis for Hierarchical Video Shot Classification. *Proc. IEEE International Conference on Image Processing*, vol. 2, pp. 550–555, 1997.
- [Navarro01] Navarro, G. A Guided Tour to Approximate String Matching. *ACM Computing Surveys*, vol. 33, no. 1, pp. 31–88, March, 2001.
- [Nucibella05] Nucibella, F., Porcelluzzi, S., and Zattra, L. Computer Music Analysis via a Multidisciplinary Approach. *Sound and Music Computing*, Salerno, Italy, November 2005.
- [Ong04] Ong, B. and Herrera, P. Computing Structural Descriptions of Music through the Identification of Representative Excerpts from Audio Files, *Proceedings of 25th International AES Conference London*, UK, June 2004.
- [Otsu79] Otsu, N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics (SMC)*, vol. 9, no. 1, pp. 62–66, 1979.
- [Peeters02] Peeters, G., Laburthe, A., and Rodet, X. Toward Automatic Music Audio Summary Generation From Signal Analysis. *International Conference on Music Information Retrieval, ISMIR*, Paris, France, 2002.
- [Pfeiffer01] Pfeiffer, S., Lienhart, R., and Effelsberg, W. Scene Determination Based on Video and Audio Features. *Multimedia Tools and Applications*, vol. 15, no. 1, pp. 59–81, 2001.
- [Povel85] Povel, D.-J., and Essens, P. Perception of Temporal Patterns. *Music Perception*, vol. 2, pp. 411–440, 1985.
- [Purwins00] Purwins, H., Blankertz, B., and Obermayer, K. A New Method for Tracking Modulations in Tonal Music in Audio Data Format. In *International Joint Conference on Neural Network (IJCNN'00)*, vol. 6, pp. 270–275, 2000.
- [Puzicha99] Puzicha, J., Hofmann, T., and Buhmann, J.M. Histogram Clustering for Unsupervised Image Segmentation. *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, Fort Collins, 1999.
- [Rabiner86] Rabiner, L. R. and Juang, B. H. An Introduction to Hidden Markov Models. *IEEE ASSP Magazine*, pp. 4–15, 1986.
- [Rabiner89] Rabiner, L. R. A tutorial on hidden Markov Models and Selected Applications In Speech Recognition. *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, February 1989.
- [Rabiner93] Rabiner L. and Juang, B. H. *Fundamentals of Speech Recognition*. Prentice-

- Hall, 1993.
- [Raphael02] Raphael, C. Automatic Transcription of Piano Music. *International Conference on Music Information Retrieval (ISMIR)*, Paris, France, 2002.
- [Rentfrow03] Rentfrow, P. J. and Gosling, S. D. The Do Re Mi's of Everyday Life: The Structure and Personality Correlates of Music Preferences. *Journal of Personality and Social Psychology*, vol. 84, no. 6, pp. 1236–1256, 2003.
- [Roediger05] Roediger, H. L. Memory (psychology). *Microsoft® Encarta® Online Encyclopedia 2005* <http://encarta.msn.com>. 2005.
- [Rosenthal92] Rosenthal, D. Emulation of Human Rhythm Perception. *Computer Music Journal*, vol. 16, no. 19, pp. 64-76, 1992.
- [Sandvold05] Sandvold, V. and Herrera, P. Towards a Semantic Descriptor of Subjective Intensity in Music. *Proceedings of International Computer Music Conference*, Barcelona, 2005.
- [Schellenberg99] Schellenberg, E. G., Iverson, P., and McKinnon, M.C. Name That Tune: Identifying Popular Recordings from Brief Excerpts. *Psychonomic Bulletin & Review*, vol. 6, pp. 641-646, 1999.
- [Selfridge97] Selfridge-Field, E. (editor), *Beyond MIDI: The Handbook of Musical Codes*. Cambridge, Massachusetts: MIT Press, 1997.
- [Selfridge98] Selfridge-Field, E. Conceptual and Representational Issues in Melodic Comparison. *Melodic Comparison: Concepts, Procedures, and Applications, Computing in Musicology*, vol. 11, pp. 3-64, 1998.
- [Siegler97] Siegler, M. A., Jain, U., Raj, B., and Stern, R. M. Automatic Segmentation, Classification and Clustering of Broadcast News Audio. *Proceeding of the DARPA Speech Recognition Workshop*, pp. 97-99, 1997.
- [Shao05] Shao, X., Maddage, N., Xu, C., and Kankanhalli, M. Automatic Music Summarization Based on Music Structure Analysis. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP05)*, vol. 2, pp. 1169-1172, 2005.
- [Shum94] Shum, M. The Role of Temporal Landmarks in Autobiographical Memory Processes. *Psychological Bulletin*, vol. 124, pp. 423-442, 1994.
- [Smith04] Smith, M. A. and Kanade, T. *Multimodal Video Characterization and Summarization (The Kluwer International Series in Video Computing)*, Norwell, MA: Kluwer Academic Publishers, 2004.
- [Smith78] Smith, S. M., Glenberg, A., and Bjork, R. A. Environmental Context and Human Memory. *Memory & Cognition*, vol. 6, no. 4, pp. 342-353, 1978.
- [Smith79] Smith, S. M. Remembering In and Out of Context. *Journal of Experimental Psychology: Human Learning and Memory*, vol. 5, pp. 460-471, 1979.

- [Solomon97] Solomon, L. *Music Theory Glossary*. Web publication, last updated 2002, <http://solo1.home.mindspring.com/glossary.htm>, 1997.
- [Steelant02] Van Steelant, D., DeBaets, B., DeMeyer, H., Leman, M., Martens, S.-P., Clarisse, L., and Lesaffre, M. Discovering Structure and Repetition in Musical Audio. In *Eurofuse*, Varanna, Italy, 2002.
- [Streich06] Streich, S. and Herrera, P. Algorithmic Prediction of Music Complexity Judgements. *9th International Conference on Music Perception and Cognition*; Bologna, Italy, 2006.
- [Sundberg77] Sundberg, J. The Acoustics of the Singing Voice. *Scientific American*, pp. 82-91, March, 1977.
- [Temperley99] Temperley, D., and Sleator, D. Modeling Meter and Harmony: A Preference Rule Approach. *Computer Music Journal*, vol. 23, no. 1, pp. 10-27, 1999.
- [Temperley01] Temperley, D. *The Cognition of Basic Music Structures*. Cambridge, MA: MIT Press. 2001.
- [Tenney80] Tenney, J., and Polanksy, L. Temporal Gestalt Perception in Music. *Journal of Music Theory*, vol. 24, pp. 205-241, 1980.
- [Thompson92] Thompson, W. F. & Cuddy, L. L. Perceived Key Movement in Four Voice Harmony and Single Voice. *Music Perception*, vol. 9, pp. 427-438, 1992.
- [Tzanetakis99] Tzanetakis, G. and Cook, P. Multifeature Audio Segmentation for Browsing and Annotation. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, New York, USA, 1999.
- [Tzanetakis02] Tzanetakis, G. Pitch Histograms in Audio and Symbolic Music Information Retrieval. *International Symposium on Music Information Retrieval*, 2002.
- [Tzanetakis04] Tzanetakis, G., Gao, J., and Steenkiste, P. A Scalable Peer-to-Peer System for Music Content and Information Retrieval. *Computer Music Journal*, vol. 28, no. 2, pp. 24-33, 2004.
- [Viterbi67] Viterbi, A. J. Error Bounds For Convolutional Codes And An Asymptotically Optimum Decoding Algorithm. *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.
- [Vos96] Vos, P. G. and Van Geenen, E. W. A Parallel-Processing Key Finding Model. *Music Perception*, vol. 14, pp. 185-224. 1996.
- [Wang00] Wang, Y., Liu, Z., and Huang J.-C. Multimedia Content Analysis Using Both Audio and Visual Clues. *IEEE Signal Processing Magazine*, pp. 12-36, 2000.
- [Warren03] Warren, J. D., Uppenkamp, S., Patterson, R. D., and Griffiths, T. D. Separating Pitch Chroma and Pitch Height in the Human Brain. *Proceedings of the National Academy of Sciences of the United States of America*, vol. 17, pp. 10038–10042, 2003.

- [Weyde03] Weyde, T. Case Study: Representation of Musical Structure for Music Software. *Proceedings of the Music Notation Workshop: XML Based Music Notation solutions*, 2003.
- [Widmer03] Widmer, G., Dixon, S., Goebel, W., Pampalk, E., and Tobudic, A. *In Search of the Horowitz Factor*. AI Magazine vol. 24, no. 3, pp. 111-130, 2003.
- [Winograd68] Winograd T. Linguistics And The Computer Analysis Of Tonal Harmony. *Journal of Music Theory*, vol. 12, pp. 2-49, 1968.
- [Xu02] Xu, C., Zhu, Y., and Tian, Q. Automatic Music Summarization Based on Temporal, Spectral and Cepstral Features. *Proceedings of IEEE International Conference on Multimedia and Expo*, pp. 117-120, 2002.
- [Young02] Young, N. *Mathematical Morphological*. <http://www.bath.ac.uk/eleceng/pages/sipg/research/morphology.htm>. Last updated July 2002.
- [Zhu03] Zhu, Y. and Zhou, D. Scene Change Detection Based on Audio and Video Content Analysis. *Proceedings of the Fifth International Conference on Computational Intelligence and Multimedia Applications (ICCIMA'03)*, 2003.

Appendix A

Glossary

This section is a glossary of the basic terminology used in this thesis provided for quick reference.

Beat: a rhythmic sub division of music usually felt as the regular timing within a piece of music.

Bridge: an interlude that connects two parts of a song and builds a harmonic connection between those parts.

Chorus (or refrain): the part of a song where a soloist is joined by a group of singers.

Clustering: the process of organizing objects into groups whose members are similar in some way.

Intro: introduction of a song.

Key-frames: the excerpts which best represent the content of a music sequence in an abstract manner, and are extracted from the original audio signal.

Onset: the change points in musical signals which are equivalent to the human perception of a new note starting.

Outro: the ending of a song

Summarization: the process of generating a short abstract from the original audio signal to represent the whole file.

Thumbnailing: the process of extracting a short excerpt from the original audio signal to represent the whole file.

Verse: the song sections that roughly corresponds to a poetic stanza. It is often sharply contrasted with the chorus (or refrain) melodically, rhythmically, and harmonically.

Appendix B

Details on Audio Database used in Chapter 3

A. 54 songs from The Beatles (1962 – 1965)

No	Title	Artist	Language
1	A Hard Day's Night	The Beatles	English
2	I Saw Her Standing There	The Beatles	English
3	It Won't Be Long	The Beatles	English
4	No Reply	The Beatles	English
5	All I've Got To Do	The Beatles	English
6	I Should Have Known Better	The Beatles	English
7	I'm A Loser	The Beatles	English
8	Misery	The Beatles	English
9	All My Loving	The Beatles	English
10	Anna Go To Him	The Beatles	English
11	Baby's In Black	The Beatles	English
12	If I Fell	The Beatles	English
13	Chains	The Beatles	English
14	Don't Bother Me	The Beatles	English
15	I'm Happy Just To Dance With You	The Beatles	English
16	I Need You	The Beatles	English
17	Rock And Roll Music	The Beatles	English
18	Boys	The Beatles	English
19	I'll Follow The Sun	The Beatles	English
20	Little Child	The Beatles	English
21	Ask Me Why	The Beatles	English
22	Mr. Moonlight	The Beatles	English
23	Tell Me Why	The Beatles	English
24	Till There Was You	The Beatles	English
25	Can't Buy Me Love	The Beatles	English

26	Kansas City Hey Hey Hey Hey	The Beatles	English
27	Please Mister Postman	The Beatles	English
28	Please Please Me	The Beatles	English
29	Any Time At All	The Beatles	English
30	Eight Days A Week	The Beatles	English
31	Roll Over Beethoven	The Beatles	English
32	Hold Me Tight	The Beatles	English
33	I'll Cry Instead	The Beatles	English
34	P. S. I Love You	The Beatles	English
35	Words Of Love	The Beatles	English
36	Baby It's You	The Beatles	English
37	Honey Don't	The Beatles	English
38	Things We Said Today	The Beatles	English
39	You Really Got A Hold On Me	The Beatles	English
40	Do You Want To Know A Secret	The Beatles	English
41	Every Little Thing	The Beatles	English
42	I Wanna Be Your Man	The Beatles	English
43	When I Get Home	The Beatles	English
44	A Taste Of Honey	The Beatles	English
45	Devil In Her Heart	The Beatles	English
46	I Don't Want To Spoil The Party	The Beatles	English
47	You Can't Do That	The Beatles	English
48	I'll Be Back	The Beatles	English
49	Not A Second Time	The Beatles	English
50	There's A Place	The Beatles	English
51	What You're Doing	The Beatles	English
52	Everybody's Trying To Be My Baby	The Beatles	English
53	Money	The Beatles	English
54	Twist And Shout	The Beatles	English

B. 27 pop songs from the Magnatune database

No	Title	Artist	Language
1	Can I Be A Star	Burnshee Thornside	English
2	I'll Be Here Awake	Arthur Yoria	English
3	Making Me Nervous	Brad Sucks	English
4	Mercurial Girl	Fluid	English
5	Unknown	Emma's Mini	English
6	What's Inside	Grayson Wray	English
7	For Madmen Only	Atomic Opera	English
8	Lamborghino	Burnshee Thornside	English
9	5 Star Fall	Fluid	English
10	Blue Glove	Emma's Mini	English
11	In 1671	Grayson Wray	English
12	Leave Me	Hybris	English
13	Headphones	Fluid	English
14	It's An Easy Life	Magnatune Compilation	English
15	The Gift	William Brooks	English
16	What I Did On My Summer Vacation	Magnatune Compilation	English
17	There You Were	Grayson Wray	English
18	Try It Like This	William Brooks	English
19	Uncommon Eloquence	Shane Jackman	English
20	Drops That Hit The Sand	Tom Paul	English
21	She's The Girl	Grayson Wray	English
22	I Didn't Catch What You Said	Tom Paul	English
23	Into The Unknown	Grayson Wray	English
24	A Different State Of Mind	William Brooks	English
25	My Heart Still Beats	Shane Jackman	English
26	Is There Anybody There	William Brooks	English
27	The Best In Me	Tom Paul	English

Appendix C

Details on Audio Database used in Chapter 4

A. BeatlesMusic - 56 songs from The Beatles 70' album

No	Title	Artist	Language
1	A Hard Day's Night	The Beatles	English
2	I Saw Her Standing There	The Beatles	English
3	It Won't Be Long	The Beatles	English
4	No Reply	The Beatles	English
5	All I've Got To Do	The Beatles	English
6	I Should Have Known Better	The Beatles	English
7	I'm A Loser	The Beatles	English
8	Misery	The Beatles	English
9	All My Loving	The Beatles	English
10	Anna Go To Him	The Beatles	English
11	Baby's In Black	The Beatles	English
12	If I Fell	The Beatles	English
13	Chains	The Beatles	English
14	Don't Bother Me	The Beatles	English
15	I'm Happy Just To Dance With You	The Beatles	English
16	I Need You	The Beatles	English
17	Rock And Roll Music	The Beatles	English
18	And I Love Her	The Beatles	English
19	Boys	The Beatles	English
20	I'll Follow The Sun	The Beatles	English
21	Little Child	The Beatles	English
22	Ask Me Why	The Beatles	English
23	Mr. Moonlight	The Beatles	English
24	Tell Me Why	The Beatles	English
25	Till There Was You	The Beatles	English
26	Can't Buy Me Love	The Beatles	English

27	Kansas City Hey Hey Hey Hey	The Beatles	English
28	Please Mister Postman	The Beatles	English
29	Please Please Me	The Beatles	English
30	Any Time At All	The Beatles	English
31	Eight Days A Week	The Beatles	English
32	Love Me Do	The Beatles	English
33	Roll Over Beethoven	The Beatles	English
34	Hold Me Tight	The Beatles	English
35	I'll Cry Instead	The Beatles	English
36	P. S. I Love You	The Beatles	English
37	Words Of Love	The Beatles	English
38	Baby It's You	The Beatles	English
39	Honey Don't	The Beatles	English
40	Things We Said Today	The Beatles	English
41	You Really Got A Hold On Me	The Beatles	English
42	Do You Want To Know A Secret	The Beatles	English
43	Every Little Thing	The Beatles	English
44	I Wanna Be Your Man	The Beatles	English
45	When I Get Home	The Beatles	English
46	A Taste Of Honey	The Beatles	English
47	Devil In Her Heart	The Beatles	English
48	I Don't Want To Spoil The Party	The Beatles	English
49	You Can't Do That	The Beatles	English
50	I'll Be Back	The Beatles	English
51	Not A Second Time	The Beatles	English
52	There's A Place	The Beatles	English
53	What You're Doing	The Beatles	English
54	Everybody's Trying To Be My Baby	The Beatles	English
55	Money	The Beatles	English
56	Twist And Shout	The Beatles	English

B. ChaiMusic - 26 songs by The Beatles from the years 1962-1966

No	Title	Artist	Language
1	A Hard Day's Night	The Beatles	English
2	Day Tripper	The Beatles	English
3	Drive My Car	The Beatles	English
4	Help	The Beatles	English
5	Eleanor Rigby	The Beatles	English
6	From Me To You	The Beatles	English
7	Norwegian Wood	The Beatles	English
8	We Can Work It Out	The Beatles	English
9	All My Loving	The Beatles	English
10	Paperback Writer	The Beatles	English
11	You've Got To Hide Your Love Away	The Beatles	English
12	Nowhere Man	The Beatles	English
13	She Loves You	The Beatles	English
14	And I Love Her	The Beatles	English
15	Yellow Submarine	The Beatles	English
16	Can't Buy Me Love	The Beatles	English
17	Michelle	The Beatles	English
18	Please Please Me	The Beatles	English
19	Ticket To Ride	The Beatles	English
20	Eight Days A Week	The Beatles	English
21	Love Me Do	The Beatles	English
22	Girl	The Beatles	English
23	In My Life	The Beatles	English
24	Yesterday	The Beatles	English
25	I Feel Fine	The Beatles	English
26	I Want To Hold Your Hand	The Beatles	English

C. WordPop - 23 popular songs in various languages

No	Title	Artist	Language
1	Bye Bye Bye	'N Sync	English
2	I'm Real	Jennifer Lopez	English
3	Brown Sugar	Rolling Stones	English
4	Chain Of Fools	Aretha Franklin	English
5	Independent Women	Destiny's Child	English
6	True Love story	Seiko Matsuda	Japanese
7	My Love Grows Deeper	Nelly Furtado	English
8	At The Beginning	Richard Marx & Donna Lewis	English
9	I am Your Angel	Celine Dion R. Kelly	English
10	You Are The Inspiration	Chicago	English
11	For Whom The Bell Tolls	Bee Gees	English
12	I Say A Little Prayer	instrumental	English
13	Bilakah Cinta	Kris Dayanti	Indonesian
14	Kau Dan Aku	Kris Dayanti	Indonesian
15	Turn It Into Love	Kylie Minogue	English
16	Out Of My Head	Kylie Minogue	English
17	当年情	Leslie Cheung	Cantonese
18	Room In Your Heart	Mike Francis	English
19	月亮代表我的心	Teresa Teng	Mandarin
20	排球女将	Seiko Matsuda	Japanese
21	Sir Duke	Stevie Wonder	English
22	Jump	Van Halen	English
23	红颜白发	Leslie Cheung	Cantonese

Appendix D

Details on Audio Database used in Chapter 5

A. 18 popular songs from The Beatles' and other artists or groups

No	Title	Artist	Language
1	Bye Bye Bye	'N Sync	English
2	Brown Sugar	Rolling Stones	English
3	Chain Of Fools	Aretha Franklin	English
4	Independent Women Part.1	Destiny Child	English
5	I Am Your Angel	Celine Dion & R. Kelly	English
6	I Believe In You And Me	Whitney Houston & Mariah Carey	English
7	I Can Wait Forever	Air Supply	English
8	Room In Your Heart	Mike Francis	English
9	Jump	Van Halen	English
10	No Reply	The Beatles	English
11	Please Mister Postman	The Beatles	English
12	Do You Want To Know A Secret	The Beatles	English
13	All My Loving	The Beatles	English
14	Things We Said Today	The Beatles	English
15	Can't Buy Me Love	The Beatles	English
16	If I Fell	The Beatles	English
17	Eight Days A Week	The Beatles	English
18	It Won't Be Long	The Beatles	English