TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY
Julkaisu 776 • Publication 776

Matti Ryynänen

Automatic Transcription of Pitch Content in Music and Selected Applications

Tampere 2008

Tampereen teknillinen yliopisto. Julkaisu 776
Tampere University of Technology. Publication 776

Matti Ryynänen

Automatic Transcription of Pitch Content in Music and Selected Applications

Thesis for the degree of Doctor of Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB109, at Tampere University of Technology, on the 12th of December 2008, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2008

ISBN 978-952-15-2074-7 (printed)
ISBN 978-952-15-2118-8 (PDF)
ISSN 1459-2045

Abstract

Transcription of music refers to the analysis of a music signal in order to produce a parametric representation of the sounding notes in the signal. This is conventionally carried out by listening to a piece of music and writing down the symbols of common musical notation to represent the occurring notes in the piece. Automatic transcription of music refers to the extraction of such representations using signal-processing methods.

This thesis concerns the automatic transcription of pitched notes in musical audio and its applications. Emphasis is laid on the transcription of realistic polyphonic music, where multiple pitched and percussive instruments are sounding simultaneously. The methods included in this thesis are based on a framework which combines both low-level acoustic modeling and high-level musicological modeling. The emphasis in the acoustic modeling has been set to note events so that the methods produce discrete-pitch notes with onset times and durations as output. Such transcriptions can be efficiently represented as MIDI files, for example, and the transcriptions can be converted to common musical notation via temporal quantization of the note onsets and durations. The musicological model utilizes musical context and trained models of typical note sequences in the transcription process. Based on the framework, this thesis presents methods for generic polyphonic transcription, melody transcription, and bass line transcription. A method for chord transcription is also presented.

All the proposed methods have been extensively evaluated using realistic polyphonic music. In our evaluations with 91 half-a-minute music excerpts, the generic polyphonic transcription method correctly found 39% of all the pitched notes (recall) where 41% of the transcribed notes were correct (precision). Despite the seemingly low recognition rates in our simulations, this method was top-ranked in the polyphonic note tracking task in the international MIREX evaluation in 2007 and 2008. The methods for the melody, bass line, and chord transcription

were evaluated using hours of music, where F-measure of 51% was achieved for both melodies and bass lines. The chord transcription method was evaluated using the first eight albums by The Beatles and it produced correct frame-based labeling for about 70% of the time.

The transcriptions are not only useful as human-readable musical notation but in several other application areas too, including music information retrieval and content-based audio modification. This is demonstrated by two applications included in this thesis. The first application is a query by humming system which is capable of searching melodies similar to a user query directly from commercial music recordings. In our evaluation with a database of 427 full commercial audio recordings, the method retrieved the correct recording in the top-three list for the 58% of 159 hummed queries. The method was also top-ranked in "query by singing/humming" task in MIREX 2008 for a database of 2048 MIDI melodies and 2797 queries. The second application uses automatic melody transcription for accompaniment and vocals separation. The transcription also enables tuning the user singing to the original melody in a novel karaoke application.

Preface

This work has been carried out at the Department of Signal Processing, Tampere University of Technology, during 2004–2008. I wish to express my deepest gratitude to my supervisor Professor Anssi Klapuri for his invaluable advice, guidance, and collaboration during my thesis work. I also wish to thank my former supervisor Professor Jaakko Astola and the staff at the Department of Signal Processing for the excellent environment for research.

The Audio Research Group has provided me the most professional and relaxed working atmosphere, and I wish to thank all the members, including but not limited to Antti Eronen, Toni Heittola, Elina Helander, Marko Helén, Teemu Karjalainen, Konsta Koppinen, Teemu Korhonen, Annamaria Mesaros, Tomi Mikkonen, Jouni Paulus, Pasi Pertilä, Hanna Silen, Sakari Tervo, Juha Tuomi, and Tuomas Virtanen.

The financial support provided by Tampere Graduate School in Information Science and Engineering (TISE), Tekniikan edistämissäätiö (TES), and Keksintösäätiö (Foundation for Finnish Inventors) is gratefully acknowledged. This work was partly supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence Program 2006–2011).

I wish to thank my parents Pirkko and Reino, all my friends and band members, and naturally – music.

The ultimate thanks go to Kaisa for all the love, understanding, and support.

Matti Ryynänen
Tampere, 2008

Contents

List of Included Publications

This thesis consists of the following publications, preceded by an introduction to the research field and a summary of the publications. Parts of this thesis have been previously published and the original publications are reprinted, by permission, from the respective copyright holders. The publications are referred to in the text by notation [P1], [P2], and so forth.

P1 M. Ryynänen and A. Klapuri, "Polyphonic music transcription using note event modeling," in *Proceedings of the 2005 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, (New Paltz, New York, USA), pp. 319–322, Oct. 2005.

P2 M. Ryynänen and A. Klapuri, "Transcription of the singing melody in polyphonic music," in *Proceedings of the 7th International Conference on Music Information Retrieval*, (Victoria, Canada), pp. 222–227, Oct. 2006.

P3 M. Ryynänen and A. Klapuri, "Automatic bass line transcription from streaming polyphonic audio," in *Proceedings of the 2007 IEEE International Conference on Acoustics, Speech, and Signal Processing*, (Honolulu, Hawaii, USA), pp. 1437–1440, Apr. 2007.

P4 M. Ryynänen and A. Klapuri, "Automatic transcription of melody, bass line, and chords in polyphonic music," *Computer Music Journal*, 32:3, pp. 72–86, Fall 2008.

P5 M. Ryynänen and A. Klapuri, "Query by humming of MIDI and audio using locality sensitive hashing," in *Proceedings of the 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing*, (Las Vegas, Nevada, USA), pp. 2249–2252, Apr. 2008.

P6 M. Ryynänen, T. Virtanen, J. Paulus, and A. Klapuri, "Accompaniment separation and karaoke application based on automatic

melody transcription," in *Proceedings of the 2008 IEEE International Conference on Multimedia and Expo*, (Hannover, Germany), pp. 1417–1420, June 2008.

Publications [P1]–[P5] were done in collaboration with Anssi Klapuri who assisted in the mathematical formulation of the models. The implementations, evaluations, and most of the writing work was carried out by the author.

Publication [P6] was done in collaboration with Tuomas Virtanen, Jouni Paulus, and Anssi Klapuri. The original idea for the karaoke application was suggested by Jouni Paulus. The signal separation part of the method was developed and implemented by Tuomas Virtanen. The melody transcription method, application integration, evaluation, and most of the writing work was done by the author.

List of Abbreviations

BPM	Beats-per-minute
F0	Fundamental frequency
GMM	Gaussian mixture model
GUI	Graphical user interface
HMM	Hidden Markov model
LSH	Locality sensitive hashing
MFCC	Mel-frequency cepstral coefficient
MIDI	Musical Instrument Digital Interface
MIR	Music information retrieval
MIREX	Music Information Retrieval Evaluation eXchange
MPEG	Moving Picture Experts Group
QBH	Query by humming
SVM	Support vector machine
VMM	Variable-order Markov model

Chapter 1

Introduction

Transcription of music refers to the analysis of a music signal in order to produce a parametric representation of the sounding notes in the signal. Analogous to speech, which can be represented with characters and symbols to form written text, music can be represented using musical notation. The common musical notation, which is widely used to represent Western music, has remained rather unchanged for several centuries and is of great importance as a medium to convey music. Conventionally, music transcription is carried out by listening to a piece of music and writing down the notes manually. However, this is time-consuming and requires musical training.

Automatic transcription of music refers to the extraction of such representations using signal-processing methods. This is useful as such, since it provides an easy way of obtaining descriptions of music signals so that musicians and hobbyists can play them. In addition, transcription methods enable or facilitate a wide variety of other applications, including music analysis, music information retrieval (MIR) from large music databases, content-based audio processing, and interactive music systems. Although automatic music transcription is very challenging, the methods presented in this thesis demonstrate the feasibility of the task along with two resulting applications.

1.1 Terminology

The terminology used throughout the thesis is briefly introduced in the following.

Musical Sounds

Humans are extremely capable of processing musical sounds and larger musical structures, such as the melody or rhythm, without any formal musical education. *Music psychology* studies this organization of musical information into perceptual structures and the listeners' affection evoked by a musical stimulus [20]. *Psychoacoustics* studies the relationship between an acoustic signal and the percept it evokes [111]. A musical sound has four basic perceptual attributes: pitch, loudness, duration, and timbre. Pitch and timbre are only briefly introduced in the following, and for an elaborate discussion on sounds and their perception, see [106].

Pitch allows the ordering of sounds on a frequency-related scale extending from low to high, and "a sound has a certain pitch if it can be reliably matched by adjusting the frequency of a sine wave of arbitrary amplitude" [47, p. 3493]. Whereas pitch is a perceptual attribute, *fundamental frequency (F0)* refers to the corresponding physical term, measured in Hertz (Hz), and it is defined only for periodic or almost periodic signals. In this thesis, the terms pitch and fundamental frequency are used as synonyms despite their conceptual difference. Pitched musical sounds usually consist of several frequency components. For a perfectly harmonic sound with fundamental frequency f_0, the sound has frequency components at the integer multiples of the fundamental frequency, $kf_0, k \geq 1$, called *harmonics*.

Timbre allows listeners to distinguish musical sounds which have the same pitch, loudness, and duration. Timbre of a musical sound is affected by the spectral content and its temporal evolvement. More informally, the term timbre is used to denote the color or quality of a sound [106].

The perception of *rhythm* is described by *grouping* and *meter* [69]. Grouping refers to the hierarchical segmentation of a music signal into variable-sized rhythmic structures, extending from groups of a few notes to musical phrases and parts. Meter refers to a regular alteration of strong and weak beats sensed by a listener. The pulses, or beats, do not have to be explicitly spelled out in music, but they may be inducted by observing the underlying rhythmic periodicities in music. *Tempo* defines the rate of the perceptually most prominent pulse and is usually expressed as beats-per-minute (BPM).

Monophonic music refers here to music where only a single pitched sound is played at a time. In *polyphonic* music, several pitched and unpitched sounds may occur simultaneously. *Monaural* (commonly mono)

refers to single-channel audio signals whereas *stereophonic* (stereo) refers to two-channel audio signals. Commercial music is usually distributed as stereophonic audio signals.

About Music Theory and Notation

The fundamental building block of music is a *note* which is here defined by a discrete pitch, a starting time, and a duration. An *interval* refers to the pitch ratio of two notes. In particular, the interval with pitch ratio $1 : 2$ is called an *octave* which is divided into twelve notes in Western music. This results in ratio $1 : 2^{1/12}$ between adjacent note pitches, which is called a *semitone*.

The seven note pitches corresponding to the white keys of piano in one octave range are named with letters C, D, E, F, G, A, and B. The octave of the pitch can be marked after the letter, e.g., A4. *Pitch classes* express the octave equivalence of notes, i.e., notes separated by an octave (or several octaves) are from the same pitch class (e.g., C is the pitch class of C3, C4, and C5). Pitch can be modified with *accidentals*, e.g., with sharp $\sharp$ ($+1$ semitone) and flat $\flat$ (-1 semitone).

For computers, a convenient way to express note pitches is to use *MIDI*[1] *note numbers*. The MIDI note number is defined for a note with a fundamental frequency f_0 (Hz) by

$$\text{MIDI note number} = 69 + 12 \log_2 \left(\frac{f_0}{440} \right), \tag{1.1}$$

where 69 and 440 (Hz), according to the widely adopted standard tuning, correspond to the MIDI note number and to the fundamental frequency of the note A4. Equation (1.1) provides a musically convenient way of representing arbitrary F0 values in semitone units when the corresponding MIDI note numbers are not rounded to integers.

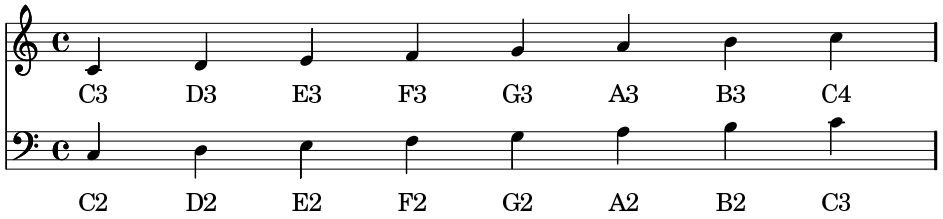
Figure 1.1 exemplifies different representations for note pitches. The tabulation in 1.1a lists note names and their fundamental frequencies in 440 Hz tuning, and the corresponding MIDI note numbers by Eq. (1.1). In 1.1b, note pitches are written with the common musical notation. The white piano keys in 1.1c correspond to the note names.

Scales are ordered series of intervals and they are commonly used as the tonal framework for music pieces. In Western music, the most commonly used scale is the seven-note *diatonic* scale which consists

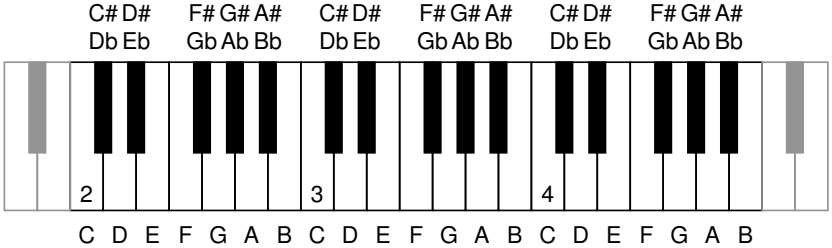
[1] Music Instrument Digital Interface (MIDI) is a standard format for coding note and instrument data.

Note	C3	D3	E3	F3	G3	A3	B3	C4
f_0, MIDI	48	50	52	53	55	57	59	60
f_0, Hz	130.8	146.8	164.8	174.6	196.0	220.0	246.9	261.6

(a) Fundamental frequencies and MIDI note numbers for note names.



(b) Note pitches on the common musical notation.



(c) Note pitches on a piano keyboard. The numbers 2, 3, and 4 identify the octave for note pitches ranging from C to B.

Figure 1.1: Different representations for note pitches.

of an interval series 2, 2, 1, 2, 2, 2, 1 semitones. The term *tonic note* refers to the note from which the interval series is started. Starting such a series from note C results in stepping through the white piano keys and C *major* scale, named after the tonic. Starting from note A with ordering 2, 1, 2, 2, 1, 2, 2 also steps through the white keys but results in A *minor* scale. Since the diatonic scale can be started from seven different positions, there are seven *modes* for the diatonic scale of which the most commonly used ones are the major (Ionian) and the minor (Aelion).

A *chord* is a combination of notes which sound simultaneously or nearly simultaneously, and three-note chords are called *triads*. Chord progressions largely determine the tonality, or harmony, of a music piece. Musical *key* identifies the tonic triad (major or minor) which represents the final point of rest for a piece, or the focal point of a section. If the major and minor modes contain the same notes, the corresponding keys are referred to as the *relative keys* (or the relative-key pair). For example, C major and A minor keys form a relative-key pair. *Key*

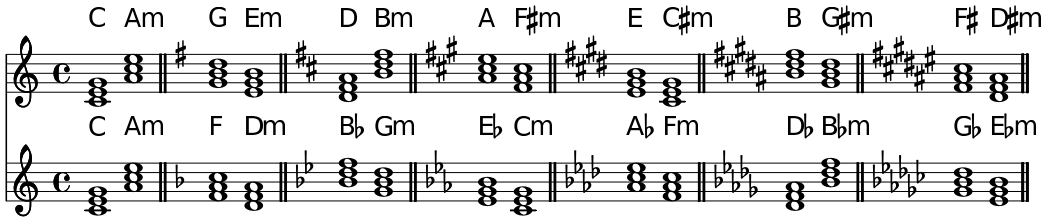


Figure 1.2: Key signatures and their tonic triads for major and minor modes.

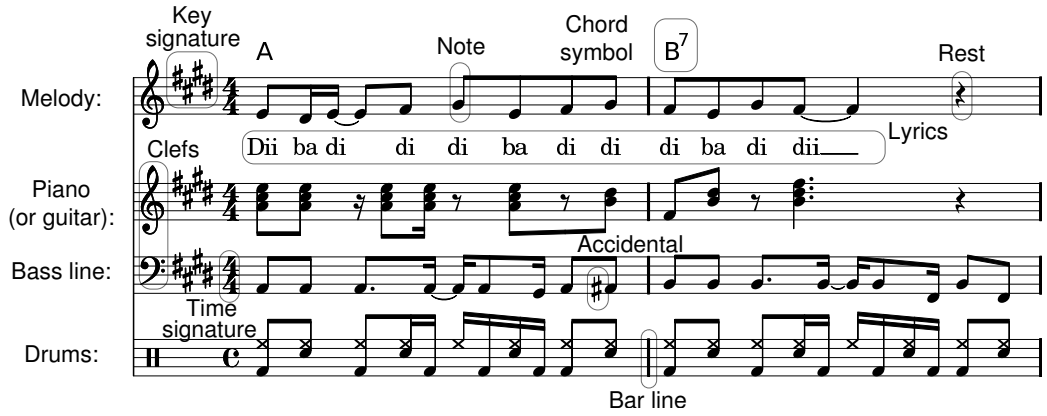


Figure 1.3: An example of common musical notation.

signature refers to a set of sharps or flats in the common musical notation defining the notes which are to be played one semitone lower or higher, and its purpose is to minimize the need for writing the possible accidentals for each note. Figure 1.2 shows the key signatures together with their major and minor tonic triads.

Figure 1.3 shows an example of the common musical notation. It consists of the note and rest symbols which are written on a five-line staff, read from left to right. The pitch of a note is indicated by the vertical placement of the symbol on the staff, possibly modified by accidentals. Note durations are specified by their stems or note-head symbols and they can be modified with dots and ties. *Rests* are pauses when there are no notes to be played. The staff usually begins with a clef, which specifies the pitches on the staff, and a key signature. Then, a time signature defines the temporal grouping of music into measures, or bars. In Figure 1.3, for example, the time signature 4/4 means that a measure lasts four quarter notes. Each measure is filled up with notes and rests so that their non-overlapping durations sum up to the length of the measure. This determines the starting point for each note or

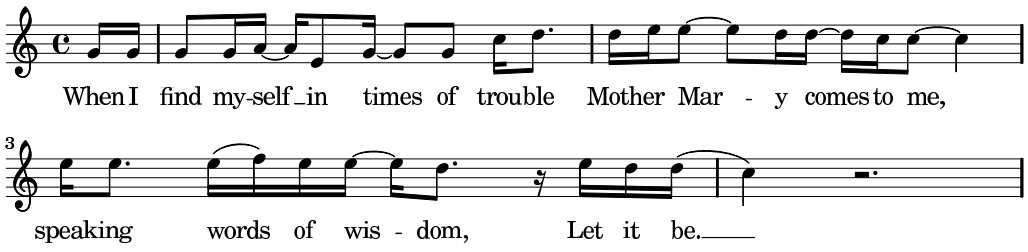
rest. Chord symbols are commonly used as a short-hand notation instead of explicitly writing all the sounding notes on the staff. Lyrics of the music piece may be printed within the notation. The notation may also include numerous performance instructions, such as tempo changes, dynamic variations, playing style, and so on. In addition to pitched instruments, percussive instruments such as drums can be notated with a certain set of symbols.

The example in Figure 1.3 shows notation for the melody, bass line, and accompaniment. *Melody* is an organized sequence of consecutive notes and rests, usually performed by a lead singer or by a solo instrument. More informally, the melody is the part one often hums along when listening to a music piece. The *bass line* consists of notes in a lower pitch register and is usually played with a bass guitar, a double bass, or a bass synthesizer. The term *accompaniment* refers to music without the melody, and in this example, it consists of the piano or guitar chords, bass line, and drums.

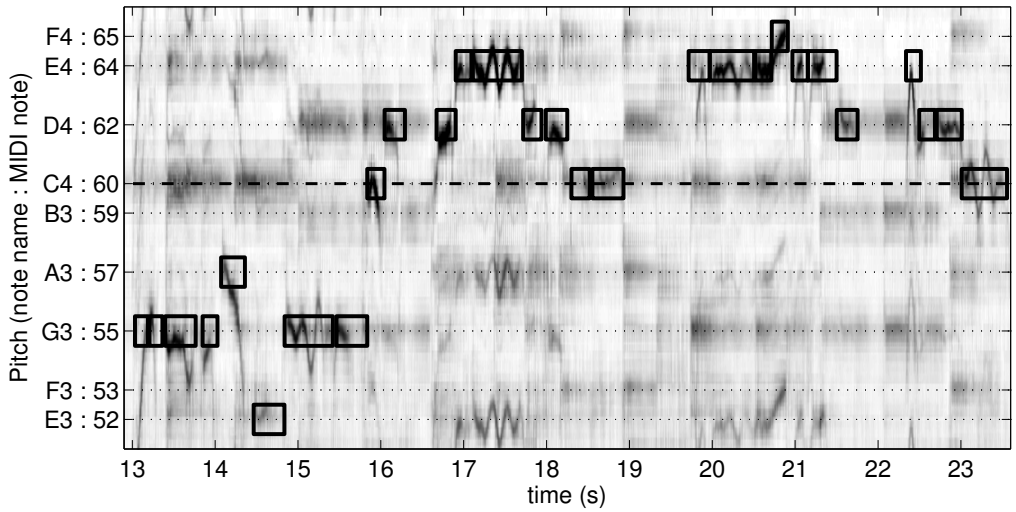
In addition to the common musical notation, MIDI files are widely used as a parametric representation of music with computers. MIDI files are very compact and flexible with wide application support. For example, MIDI is the standard for representing notes and other control data in computer music software and hardware; it is used in sequencers, music notation software, music-controlled effects and lighting, and even in ring tones for mobile phones.

A common visualization of MIDI notes is called a *piano roll*. As an example, Figure 1.4 illustrates the measures of the melody from song "Let It Be" by The Beatles. Panel 1.4a shows the melody and lyrics in music notation. In panel 1.4b, the black rectangles show the melody notes with the piano-roll representation where time and pitch are on horizontal and vertical axes, respectively. The note pitches in both representations are discrete. In the piano roll, however, the note starting times and durations are not discrete but continuous in time. For illustration purposes, the strength of fundamental frequencies, estimated from the original recording by The Beatles, are indicated by the gray-level intensity in the background. The horizontal dashed lines denote the note pitches of C major key.

In this example, the musical notation and the piano-roll representation match each other quite accurately. The plotted strengths of fundamental frequencies, however, reveal the use of vibrato and glissandi in the singing performance. For example, typical examples of vibrato occur at 15 s ("times"), 17 s ("Mar-"), and 23 s ("be") whereas glissandi occur at 13 s ("When"), 14.2 s ("-self"), 19.8 s ("speaking"), and 22.4 s



(a) Melody notes and lyrics written with music notation.



(b) Melody notes (the black rectangles) shown with piano-roll representation. In the background, the gray-level intensity shows the strength of fundamental frequencies estimated from the original recording for illustration.

Figure 1.4: The first melody phrase of "Let It Be" by The Beatles in music notation and piano-roll representation.

("Let"). The example also illustrates the ambiguity in music transcription which makes the transcription task challenging for machines: although the transcription is obviously correct for human listeners, the fundamental frequencies vary widely around the discrete note pitches. As an example, the fundamental frequencies span over five semitones during the note "Let" (22.4 s) with glissando and only briefly hit the discrete note E4 in the transcription. In addition, the discrete note durations in the musical notation are often quantized to longer durations to make the notation easier to read. For example, the note at 18.5 s ("me") is performed as an eighth-note whereas in the musical notation the same note is extended to the end of the measure. To summarize, musical notations provide guidelines for musicians to reproduce musi-

cal pieces, however, with the freedom to produce unique music performances.

1.2 Overview of Automatic Music Transcription

As already mentioned, humans are extremely good at listening to music. The ability to focus on a single instrument at a time, provided that it is somewhat audible in the mixture, is especially useful in music transcription. The usual process of manual music transcription proceeds in top-down order: first, the piece is segmented into meaningful parts and the rhythmic structure is recognized. After that, the instruments or musical objects of interest (e.g., the singing melody or chords) are written down by repeatedly listening to the piece. Most of the automatic transcription methods in the literature, however, use a bottom-up approach, including the methods proposed in this thesis. This is due to the fact that modeling such analytic listening and organization of musical sounds into entities is simply a very challenging problem.

At a lower level, *relativity* is a fundamental difference in the perception of musical objects between humans and machines. Musically trained subjects can distinguish musical intervals and rhythmic structures relative to tempo. However, only a few people can actually directly name the absolute pitch of a sounding note, or the tempo of the piece. Therefore, musicians commonly use an instrument to determine the note names while doing the transcription. With transcription methods, however, the absolute values can be measured from the input signal.

Research Areas and Topics

There exists a wide variety of different research topics in music-signal processing concerning the analysis, synthesis, and modification of music signals. In general, automatic music transcription tries to extract information on the musical content and includes several topics, such as pitch and multipitch estimation, the transcription of pitched instruments, sound-source separation, beat tracking and meter analysis, the transcription of percussive instruments, instrument recognition, harmonic analysis, and music structure analysis. These are briefly introduced in the following, with some references to the relevant work.

In order to build an ultimate music transcription system, all of these are important. For a more complete overview of different topics, see [63]. An international evaluation festival, Music Information Retrieval Evaluation eXchange (MIREX), nicely reflects the hot-topic tasks in music-signal processing. See [24] for an overview of MIREX and [1, 2] for 2007–2008 the results and abstracts.

Pitch and multipitch estimation of music signals is a widely studied task and it is usually a prerequisite for pitched instrument transcription. The usual aim is to estimate one or more fundamental frequency values within a short time frame of the input signal, or judge the frame to be unvoiced. The estimated pitches in consequent time frames are usually referred to as a *pitch track*. There exist several algorithms for fundamental frequency estimation in monophonic signals, including time-domain and frequency-domain algorithms, and it is practically a solved problem (for reviews, see [50, 18]). The estimation of several pitches in polyphonic music is a considerably more difficult problem for which, however, a number of feasible solutions have been proposed [63, 17, 62].

In general, *pitched instrument transcription* refers to the estimation of pitch tracks or notes from musical audio signals. Transcription of notes requires both note segmentation and labeling. Here the term *note segmentation* refers to deciding the start and end times for a note and the term *note labeling* to assigning a single pitch label (e.g., a note name or a discrete MIDI note number) to the note. The input can be monophonic or polyphonic and also the transcription output can be either one. In the monophonic case, singing transcription has received most attention (see [108] for an introduction and a review of methods). For polyphonic inputs, the first transcription method dates back more than thirty years [83]. Along several methods evaluated on synthetic inputs (e.g., random mixtures of acoustic samples or music signals synthesized from MIDI files), there exist methods for transcribing real-world music taken, for example, from commercial music CDs. Pitch tracking of the melody and bass lines in such material was first considered by Goto [35, 36]. Later, either pitch tracking or note-level transcription of the melody has been considered, for example, in [28, 92, 29, 25]; and [P2], [P4], and the bass line transcription in [43] and [P3], [P4]. Various melody pitch tracking methods have been evaluated in MIREX in 2005, 2006, and 2008, preceded by ISMIR 2004 audio description task. The results and methods of the 2004–2005 comparative evaluations are summarized in [98].

Methods for producing polyphonic transcriptions from music have also been under extensive research. In this context, the research has focused on the transcription of piano music, including [105, 21, 104, 74, 99, 82, 126]. However, some of these methods are also applicable to "generic" music transcription, together with methods including [58, 9, 6], and [P1]. The generic transcription task was considered in MIREX 2007–2008 with title "Multiple Fundamental Frequency Estimation & Tracking". Currently, polyphonic music transcription is not a completely solved problem, despite the encouraging results and development in recent years.

Sound-source separation aims at recovering the audio signals of different sound sources from a mixture signal. For music signals, this is a particularly interesting research area which enables, for example, the acoustic separation of different instrument sounds from complex polyphonic music signals. The method proposed in [P6] combines melody transcription and sound-source separation in order to suppress vocals in commercial music recordings. Some approaches to sound-source separation are briefly introduced in Section 7.2.

Beat tracking refers to estimating the locations of beats in music signals with possibly time-varying tempi. For humans, this is a seemingly easy task: even an untrained subject can usually sense the beats and correctly tap foot or clap hands along with a music piece. *Tempo estimation* refers to finding the average rate of the beats. *Meter analysis* of music signals aims at a more detailed analysis in order to recover the hierarchical rhythmic structure of the piece. For example, the method proposed in [64] produces three levels of metrical information as output: tatum, tactus (beats), and measures. *Onset detection* methods aim at revealing the beginning times of individual, possibly percussive, notes. For an overview and evaluation of beat tracking, tempo estimation, and meter analysis methods, see [40, 42, 77]. Tempo tracking and quantization of note timings (e.g., in MIDI files) into discrete values is a related topic [8, 134], which facilitates the conversion of automatic transcriptions in MIDI format into common musical notation.

The transcription of unpitched percussive instruments, such as the bass drum, the snare drum, and the cymbals in a typical drum set, is also an interesting research topic which, however, has not gained as much research emphasis as the pitched instrument transcription. The developed methods can be broadly categorized into pattern recognition and separation-based methods. Despite the rapid development of the methods, their performance is still somewhat limited for polyphonic music. For different approaches and results, see [30, 138].

Instrument recognition methods aim at classifying the sounding instrument, or instruments, in music signals. The methods usually model the instrument timbre via various acoustic features. Classification of isolated instrument sounds can be performed rather robustly by using conventional pattern recognition algorithms. The task is again much more complex for polyphonic music and usually the methods use sound-source separation techniques prior to classification. The transcription of percussive instruments is closely related to instrument recognition. For different approaches and results, see [49, 96, 12].

Methods for *harmonic analysis* attempt to extract information about the tonal content of music signals. Common tasks include key estimation and chord labeling which are later discussed in Chapter 5. For an extensive study, see [34]. In addition to the analysis of acoustic inputs, work has been carried out to analyze the tonal and rhythmical content in MIDI files. A particularly interesting work has been implemented as the "Melisma Music Analyzer" program[2] by Temperley and Sleator, based on the concepts presented in [119, 118].

Music structure analysis refers to extracting a high-level sectional form for a music piece, i.e., segmenting the music piece into parts and possibly assigning labels (such as "verse" and "chorus") to them. Recently, music structure analysis has become a widely studied topic with practical applications including music summarization, browsing, and retrieval [10, 88, 70, 94].

Applications

Automatic music transcription enables or facilitates several different applications. Despite the fact that the performance of the methods is still somewhat limited, the transcriptions are useful as such in music notation software for music hobbyists and musicians, and for music education and tutoring. The transcription can be corrected by the user if it is necessary to obtain a perfect transcription. Such semi-automatic transcriptions could be distributed in a community-based web service. Currently, there exist several sites in the web providing manually prepared MIDI files, guitar tablatures, and chord charts of popular songs.[3] Wikifonia is a recent example of a service providing lead sheets of music pieces prepared by its users (www.wikifonia.org).

[2] Available at www.link.cs.cmu.edu/music-analysis

[3] However, one must carefully take into account the copyright issues when distributing music in such formats.

There exist some software for automatic music transcription, although with limited performance, usually referred to as wave-to-MIDI conversion tools. These include Digital Ear (www.digital-ear.com), Solo Explorer (www.recognisoft.com), and Autoscore (www.wildcat.com) for monophonic inputs; and AKoff Music Composer (www.akoff.com), TS-AudioToMIDI (http://audioto.com), and Intelliscore (www.intelliscore.net) for polyphonic inputs. Transcribe! (www.seventhstring.com) is an example of a tool for aiding manual transcription.

Applications of music information retrieval are of great importance. The revolution in the way people consume and buy music has resulted in an extremely rapid growth of digital music collections. Within the next decade, it is expected that music is mostly sold as digital music files in online media stores.[4] Therefore, applications for browsing and retrieving music based on the musical content are important for both consumers and the service providers. *Query by humming* (QBH) is an example of music information retrieval where short audio clips of humming (e.g., the melody of the desired piece) act as queries. Automatic melody transcription is useful for producing a parametric representation of the query and of the recordings in an audio collection in the case of audio retrieval. A method for this is proposed in [P5] (see Section 7.1). There also exist query by tapping systems where the search is based on the similarity of rhythmic content [55]. Examples of web services for query by humming and query by tapping can be found at www.midomi.com and at www.melodyhound.com, respectively. Also music browsing applications may use intra-song or inter-song similarity measures based on automatic music transcription. As an example, melodic fragments can be used to search for the repeating note sequences within a music piece, or then to search for other music pieces with similar melodies.

Content-based music modification methods allow the user to perform modifications to a piece based on automatically extracted information. A beat-tracking method can be used to synchronize music pieces for remixing purposes, for example. Melodyne software is an example of content-based music modification for professional music production. It automatically transcribes the notes from a monophonic input sound and allows the user to edit the audio of individual notes, for example, by pitch shifting and time stretching. Celemony, the company behind Melodyne, has introduced a new version of the Melodyne with

[4] Apple® announced in July 2007 that over three billion songs have been purchased via their online music store iTunes®.

a feature called “direct note access”, which enables editing individual notes in polyphonic music as well. This feature will be available on the Melodyne plugin version 2 scheduled for publication in the beginning of 2009. For details, see www.celemony.com. An example application for editing individual notes in polyphonic music was also presented in [130].

Object-based coding of musical audio aims at using high-level musical objects, such as notes, as a basis for audio compression. While MIDI is a highly structured and compact representation of musical performance data, the MPEG-4 “structured audio standard“ defines a framework for representing the actual sound objects in a parametric domain [123]. The standard includes, e.g., Structured Audio Score Language (SASL) for controlling sound generation algorithms. MIDI can be used interchangeably with SASL for backward compatibility. Although the standard has existed for a decade, the object-based coding of complex polyphonic music is currently an unsolved problem. However, good audio quality with low bit rates (less than 10 kbit/s) has been achieved for polyphonic music with no percussive instruments and limited polyphony [127].

Interactive music systems can utilize automatic music transcription in various manners. For example, beat tracking can be used for controlling stage lighting or computer graphics in live performances [37]. Also the video game industry has recently demonstrated the huge market potential of games based on interaction with musical content, including titles like “Guitar Hero”, “Rockband”, “Singstar”, and “Staraoke”. *Score following and alignment* refers to systems where a musical score is synchronized either with a real-time input from the user performance or with an existing audio signal [102, 103, 22, 15]. The former enables an interactive computer accompaniment by synthesizing the score during the user performance. In MIDI domain, there exist interactive accompaniment methods, such as [13, 120], and also music-generating, or computer improvisation, methods [66, 91].

1.3 Objectives and Scope of the Thesis

The main objective of this thesis is to propose methods for the automatic transcription of pitched notes in music signals. The methods produce notes with discrete pitch (representable with integer MIDI note numbers or note names) and their non-quantized start and end times. This work applies a simple and efficient statistical framework

to automatic music transcription. In particular, the focus is set on complex polyphonic music to ensure the applicability of the methods to any music collection. There are no restrictions on the sounding instruments, music style, or maximum polyphony. Percussive sounds, such as drums, may be present in the input signals but they are not transcribed. In singing-melody transcription, lyrics are not recognized. Instrument recognition as such is not considered but some of the transcription methods are tailored for transcribing certain musical entities, such as the melody and bass line. Utilizing the timbre of different instruments is not addressed in this thesis.

The development of acoustic feature extractors, such as fundamental frequency estimators, is beyond the scope of the thesis. Instead, the proposed methods use these extractors as front-ends and aim at producing notes based on the features. The method development focuses on combining low-level acoustic modeling and high-level musicological modeling into a statistical framework for polyphonic music transcription. The acoustic models represent notes and rests and their parameters are learned from music recordings. The framework utilizes musicological context in terms of musical key and learned statistics on note sequences.

The second objective of the thesis is to demonstrate the applicability of the proposed transcription methods in end-user applications. First, this thesis includes a complete query by humming system which performs retrieval directly from music recordings, enabled by automatic melody transcription. The second application uses automatic melody transcription to suppress vocals in music recordings and to tune user singing in a karaoke application.

1.4 Main Results of the Thesis

As the main result and contribution, this thesis tackles the problem of realistic polyphonic music transcription with a simple and efficient statistical framework which combines acoustic and musicological modeling to produce MIDI notes as an output. Based on this framework, the thesis proposes the following transcription methods.

- A generic polyphonic music transcription method with state-of-the-art performance [P1].
- A melody transcription method where the framework has been tailored for singing voice [P2].

- A bass line transcription method which is capable of transcribing streaming audio [P3].
- A method for transcribing the melody, bass line, and chords to produce a song-book style representation of polyphonic music [P4].

The second main result of the thesis is to exemplify the use of a proposed melody transcription method in two practical applications. This shows that the produced transcriptions are useful and encourages other researchers to utilize automatic music transcription technology in various applications. The included applications are the following.

- A query by humming method with a novel and efficient search algorithm of melodic fragments [P5]. More importantly, the method can perform the search directly from music recordings which is enabled by the melody transcription method.
- A novel karaoke application for producing song accompaniment directly from music recordings based on automatic melody transcription [P6]. The melody transcription also enables the real-time tuning of the user singing to the original melody.

The results of each included publication are summarized in the following.

[P1] Generic Polyphonic Transcription

The publication proposes a method for producing a polyphonic transcription of arbitrary music signals using note event, rest, and musicological modeling. Polyphonic transcription is obtained by searching for several paths through the note models. In our evaluations with 91 half-a-minute music excerpts, the method correctly found 39% of all the pitched notes (recall) where 41% of the transcribed notes were correct (precision). Although the method introduces a simple approach to polyphonic transcription, the method was top-ranked in polyphonic note tracking tasks in MIREX 2007 and 2008 [1, 2].

[P2] Singing Melody Transcription in Polyphonic Music

The publication proposes a method for singing melody transcription in polyphonic music. The main contribution is to use the framework for a

particular transcription target by learning note model parameters for singing notes. A trained model for rests is introduced, and the estimation of singing pitch range and glissandi correction are applied. The method was evaluated using 96 one-minute excerpts of polyphonic music and achieved recall and precision rates of 63% and 45%, respectively.

[P3] Bass Line Transcription in Polyphonic Music

The publication proposes a method for transcribing bass lines in polyphonic music by using the framework. The main contribution is to address real-time transcription and causality issues to enable transcription of streaming audio. This includes causal key estimation, upper-F0 limit estimation, and blockwise Viterbi decoding. Also, variable-order Markov models are applied to capture the repetitive nature of bass note patterns both as pre-trained models and in an optional post-processing stage. The method achieved recall and precision rates of 63% and 59%, respectively, for 87 one-minute song excerpts.

[P4] Transcription of Melody, Bass Line, and Chords in Polyphonic Music

The publication proposes a note modeling scheme where the transcription target is contrasted with the other instrument notes and noise or silence. A simplified use of a pitch-salience function is proposed, and the method transcribes the melody, bass line, and chords. The method is capable of producing a song-book style representation of music in a computationally very efficient manner. Comparative evaluations with other methods show state-of-the-art performance.

[P5] Method for Query by Humming of MIDI and Audio

The publication proposes a query by humming method for MIDI and audio retrieval. The main contribution consists of a simple but effective representation for melodic fragments and a novel retrieval algorithm based on locality sensitive hashing. In addition, the search space is extended from MIDI-domain to audio recordings by using automatic melody transcription. Compared with previously reported results in the literature, the audio retrieval results are very promising. In our evaluation with a database of 427 full commercial audio recordings, the

method retrieved the correct recording in the top-three list for the 58% of 159 hummed queries. The method was also top-ranked in "query by singing/humming" task in MIREX 2008 [2] for a database of 2048 MIDI melodies and 2797 queries.

[P6] Accompaniment Separation and Karaoke Application

The publication proposes an application of automatic melody transcription to accompaniment versus vocals separation. In addition, a novel karaoke application is introduced where user singing can be tuned to the transcribed melody in real-time. A Finnish patent application of the technique was filed in October 2007 [110].

1.5 Organization of the Thesis

This thesis is organized as follows. Chapter 2 gives an overview of the proposed transcription methods with comparison to previous approaches. Chapter 3 briefly introduces the applied feature extraction methods which is followed by introductions to acoustic modeling and musicological modeling in Chapters 4 and 5, respectively. Chapter 6 summarizes the used evaluation criteria, databases, reported results, and refers to comparative evaluations of the methods in literature. Chapter 7 briefly introduces the two proposed applications based on an automatic melody transcription method. Chapter 8 summarizes the main conclusions of this thesis and outlines future directions for the development of transcription methods and the enabled applications.

Chapter 2

Overview of the Proposed Transcription Methods

All the proposed transcription methods included in this thesis employ a statistical framework which combines low-level acoustic modeling with high-level musicological modeling. Since the methods aim at producing notes with discrete pitch labels and their temporal segmentation, the entity to be represented with acoustic models has been chosen to be a *note event*[1]. The musicological model aims at utilizing the musical context and learned statistics of note sequences in the methods.

Figure 2.1 shows a block diagram of the framework. First, the input audio is processed with frame-wise feature extractors, for example, to estimate fundamental frequencies and their strengths in consecutive signal frames. The features are then passed to both acoustic and musicological modeling blocks. The acoustic models use pre-trained parameters to estimate the likelihoods of different note events and rests. More precisely, note events and rests are modeled using *hidden Markov models* (HMMs) for which the observation vectors are derived from the extracted features. The musicological model uses the features to estimate the key of the piece and to choose a pre-trained model for different note transitions. Statistics of note sequences are modeled with *N-grams* or *variable-order Markov models* (VMMs). After calculating the likelihoods for note events and their relationships, standard decoding methods, such as the Viterbi algorithm, can be used to resolve a sequence of notes and rests. The details of each block are introduced in the following chapters.

[1]In this work, the term note event refers to an acoustic realization of a note in a musical composition.

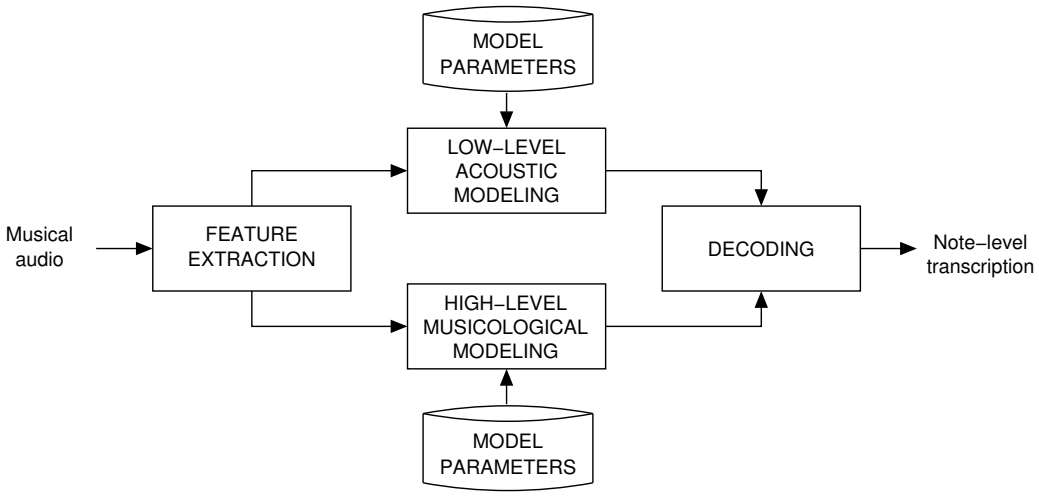


Figure 2.1: A block diagram of the framework for automatic music transcription.

The framework has several desirable properties. First, discrete pitch labels and temporal segmentation for notes are determined simultaneously. Secondly, the framework can be easily extended and adapted to handle different instruments, music style, and features by training the model parameters with the music material in demand. This is clearly demonstrated by the successful transcription methods for different transcription targets. Thirdly, the framework is conceptually simple and proves to be computationally efficient and to produce state-of-the-art transcription quality.

Table 2.1 summarizes the proposed transcription methods using the framework and lists the feature extractors and the applied techniques for low-level modeling, high-level modeling, and decoding. In publication [P1], the framework was first applied to transcribe any pitched notes to produce polyphonic transcription from arbitrary polyphonic music. After this, the framework was applied to singing melody transcription in polyphonic music [P2], and later adapted to bass line transcription of streaming audio [P3]. The transcription of the melody, bass line, and chords was considered in [P4], including streamlined performance and note modeling for target notes (i.e., melody or bass notes), other notes, and noise or silence. Details of the methods are explained in the publications.

An analogy can be drawn between the framework and large-vocabulary speech recognition systems. Hidden Markov models are conven-

Table 2.1: Summary of the proposed transcription methods.

Publ.	Target	Material	Features	Acoustic Modeling	Key Est.	Musicol. Modeling	Decoding	Post-processing
[P1]	All pitched notes	Polyphonic	Multi-F0	Note HMM and rest HMM	Yes	Bigrams	Token-passing[a], iteratively	None
[P2]	Melody	Polyphonic	Multi-F0, accent	Note HMM and rest HMM	Yes	Bigrams	Viterbi	Glissando correction
[P3]	Bass line	Polyphonic	Multi-F0, accent	Note HMM and rest HMM	Yes	Bigrams or VMM	Blockwise Viterbi	Retrain VMM + decode
[P4]	Melody or bass line	Polyphonic	Pitch salience, accent	HMMs for target notes, other notes, and noise or silence	Yes	Bigrams	Viterbi	None
[P4]	Chords	Polyphonic	Pitch salience	Chord profiles	No	Bigrams	Viterbi	None

[a]Viterbi is directly applicable. The method [P1] inherited the token-passing algorithm from our earlier work [109]. See the discussion in Section 4.1.

tionally used for modeling sub-word acoustic units or whole words and the transitions between words are modeled using a language model [51, 139]. In this sense, note events correspond to words and the musicological model to the language model as discussed in [27]. Similarly, using key information can be assimilated to utilizing context in speech recognition.

2.1 Other Approaches

Automatic transcription of the pitch content in music has been studied for over three decades resulting in numerous methods and approaches to the problem. It is difficult to properly categorize the whole gamut of transcription methods since they tend to be complex, combine different computational frameworks with various knowledge sources, and aim at producing different analysis results for different types of music material. To start with, questions to characterize a transcription method include: does the method aim at producing a monophonic or polyphonic transcription consisting of continuous pitch track(s), segmented notes, or a musical notation; what type of music material the transcription method handles (e.g., monophonic, polyphonic); what kind of a computational framework is used (e.g., rule-based, statistical, machine learning); and does the method use other knowledge sources (e.g., tone models, musicological knowledge) in addition to the acoustic input signal. Since the pioneering work of Moorer to transcribe simple duets [83], the transcription of complex polyphonic music has become the topic of interest. Examples of different approaches are provided in the following discussion.

Goto was the first to tackle the transcription of complex polyphonic music by estimating the F0 trajectories of melody and bass line on commercial music CDs [35, 36]. The method considers the signal spectrum in a short time frame as a weighted mixture of tone models. A tone model represents typical harmonic structure by Gaussian distributions centered at the integer multiples of a fundamental frequency value. Expectation-maximization algorithm is used to give maximum *a posteriori* estimate of the probability for each F0 candidate. The temporal continuity of the predominant F0 trajectory is obtained by a multiple-agent architecture. Silence is not detected but the method produces a predominant F0 estimate in each frame. The method analyzes the lower frequency range for the bass line and the middle range for the melody. Later, e.g., Marolt used analysis similar to Goto's to create sev-

eral, possibly overlapping, F0 trajectories and clustered the trajectories belonging to the melody [75]. Musicological models are not utilized in these methods. Both of them produce continuous F0 trajectories as output whereas the proposed methods produce MIDI notes.

Kashino and colleagues integrated various knowledge sources into a music transcription method [58], and first exemplified the use of probabilistic musicological modeling. They aimed at music scene analysis via hierarchical representation of frequency components, notes, and chords in music signals. Several knowledge sources were utilized, including tone memories, timbre models, chord-note relations, and chord transitions. All the knowledge sources were integrated into a dynamic Bayesian network[2]. Temporal segmentation was resolved at the chord level and results were reported for MIDI-synthesized signals with a maximum polyphony of three notes. For an overview of their work, see [57].

Bayesian approaches have been applied in signal-model based music analysis where not only the F0 but all the parameters of overtone partials are estimated for the sounding notes. Such methods include [16, 9, 127], for example. The drawback is that for complex polyphonic mixtures, the models tend to become computationally very expensive due to enormous parameter spaces.

Music transcription methods based on machine learning derive the model parameters from annotated music samples. The techniques include HMMs, neural networks, and support vector machines (SVMs), for example. Raphael used HMMs to transcribe piano music [104], where a model for a single chord consists of states for attack, sustain, and rest. The state-space for chords consists of all possible pitch combinations. At the decoding stage, however, the state-space needed to be compressed due to its huge size to contain only the most likely hypotheses of the different note combinations. The models were trained using recorded Mozart piano sonata movements.

Marolt used neural networks to transcribe piano music [74]. The method front-end used a computational auditory model followed by a network of adaptive oscillators for partial tracking. Note labeling and segmentation were obtained using neural networks. No musicological model was applied.

[2]In general, dynamic Bayesian networks model data sequences and HMMs can be considered as a special case of dynamic Bayesian networks. See [84] for a formal discussion.

Poliner and Ellis used SVMs for piano transcription [99]. Their approach was purely based on machine learning: the note-classifying SVMs were trained on labeled examples of piano music where short-time spectra acted as the inputs to the classification. The method thus made no assumptions about musical sounds, not even about the harmonic structure of a pitched note. The frame-level pitch detection was carried out by the SVM classifiers, followed by two-state on/off HMMs for each note pitch to carry out the temporal segmentation. They used similar approach to melody transcription [29] and performed well in MIREX melody transcription evaluations [98].

2.2 Earlier Methods Using Note Event Modeling

Note events have been modeled with HMMs prior to this work. For example, Raphael used two-state note HMMs with states for attack and sustain to perform score alignment [102]. The method handled monophonic music performances and required the corresponding musical score as an input.

Durey and Clements applied note HMMs for melodic word-spotting in a query by melody system [27]. A user performed the query by entering a note list as text. Then the HMMs for each note in the list were concatenated to obtain a model for the query. The model was then evaluated for each monophonic recording in the database to output a ranked list of retrieved melodies.

Shih *et al.* used three-state HMMs to model note events in monophonic humming transcription [114]. Instead of absolute pitch values, the note models accounted for intervals relative to either the first or the preceding note. In the former case, note models were trained to describe one octave of a major scale upwards and downwards from the first detected note. In the latter case, they trained models for one and two semitone intervals upwards and downwards with respect to the previous note.

Also Orio and Sette used note HMMs to transcribe monophonic singing queries [89], with states for attack, sustain, and rest. The HMMs of different notes were integrated into a note network and the Viterbi algorithm was used to decide both the note segments and the pitch labels simultaneously, thus producing note-level transcriptions. They discussed about the possibility to use the between-note transi-

tions in a musically meaningful manner, about using several attack states for modeling different types of note beginnings, and about an enhanced sustain state with two additional states to model slight detunings upwards and downwards from the note pitch. However, these ideas were not implemented in their reported system.

Viitaniemi *et al.* used a HMM, in which each state corresponded to a single MIDI note pitch, for monophonic singing transcription [124]. The transitions between the notes (i.e., each state) were controlled with a musicological model using a key estimation method and a pre-trained bigram. Viterbi algorithm was used to produce a frame-level labeling of discrete note pitches.

Our preliminary work addressed monophonic singing transcription by combining the note event HMMs with key estimation and note sequence modeling [107, 109, 108]. Although the framework itself is not a novel contribution, the methods included in this thesis demonstrate its applicability in complex polyphonic music transcription.

Chapter 3

Feature Extraction

Feature extractors are used as front-ends for the proposed transcription methods. Although no feature extractors have been developed in this thesis, this chapter briefly introduces the employed methods and summarizes the features with examples. Notice that the transcription framework is not in any way restricted to the used features but allows other extractors to be used in a straightforward manner.

3.1 Fundamental Frequency Estimators

The estimation of fundamental frequencies is important for any pitched instrument transcription system. The estimators aim at extracting a number of fundamental frequencies and their strengths, or saliences, within short time frames of an input signal. As already mentioned, F0 estimation from monophonic music has been widely studied and the problem is largely solved. One example of such an estimator is the YIN algorithm [19] which was employed, e.g., in monophonic singing transcription methods [124, 109]. For details and comparison with other approaches, see [108]. In order to transcribe polyphonic music, more complex methods are required.

The proposed transcription methods employ a number of multiple-F0 estimation algorithms by Klapuri, including [60, 61, 62]. Figure 3.1 shows an overview block diagram of these estimators. A frame of an audio signal is first converted to an intermediate representation (in the frequency domain) where the periodicity analysis takes place. In [60, 62], this transform is carried out by using a computational model of the

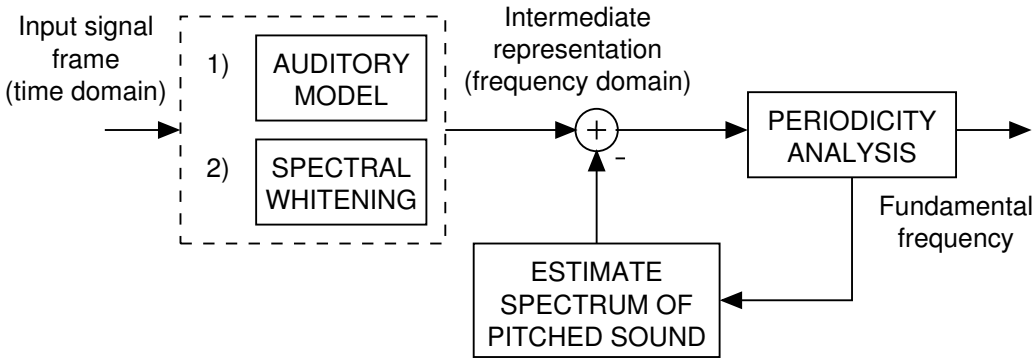


Figure 3.1: An overview of the F0 estimation methods by Klapuri.

auditory system[1]. The auditory model consists of a bandpass filterbank to model the frequency selectivity of the inner ear. Each subband signal is compressed, half-wave rectified, and lowpass filtered to model the characteristics of the inner hair cells that produce firing activity in the auditory nerve. The resulting subband signal is transformed to frequency domain. The spectra are summed over the bands to obtain the intermediate representation.

Instead of the auditory model, a computationally simpler spectral whitening can also be used to produce the intermediate representation, as proposed in [61]. Spectral whitening aims at suppressing timbral information and making the pitch estimation more robust to various sound sources. Briefly, an input signal frame is first transformed into the frequency domain. Powers within critical bands are estimated and used for calculating bandwise compression coefficients. The coefficients are linearly interpolated between the center frequencies of the bands to obtain compression coefficients for the frequency bins. The input magnitude spectrum is weighted with the compression coefficients to obtain the intermediate representation.

The periodicity analysis uses the intermediate representation to estimate the strength of each fundamental frequency candidate to produce a so-called pitch salience function. For a pitch candidate, the value of this function is calculated as a weighted sum of the amplitudes of the harmonic partials of the candidate. The global maximum of the salience function gives a good estimate of the predominant pitch in the signal frame. To obtain several pitch estimates, the spectrum of the pitched sound with the found F0 is estimated, canceled from the

[1]Some of the first models accounting for the principles of auditory processing were applied in speech processing [73] and sound quality assessment [56].

Table 3.1: Summary of the parameters for multipitch estimation.

Publication	Target	Intermediate represent.	Frame size and hop (ms)	Output	F0 region (Hz)
[P1]	All pitched notes	Auditory	92.9, 11.6	Five F0s	30–2200
[P2]	Singing melody	Auditory	92.9, 23.2	Six F0s	60–2100
[P3]	Bass line	Spectral whitening	92.9, 23.2	Four F0s	35–270
[P4]	Melody, bass line, chords	Spectral whitening	92.9, 23.2	Pitch salience function	35–1100

intermediate representation, and the search of a new F0 estimate is repeated. This iterative process is continued until a desired number of F0 estimates has been obtained.

Here, the above-described F0 estimation process has been utilized in every method for polyphonic music transcription, and the parameters are summarized in Table 3.1. The generic polyphonic transcription [P1] and melody transcription [P2] used the auditory-model based F0 estimation method with iterative F0 estimation and cancellation [62]. In [P3] for bass line transcription, the F0 estimation with spectral whitening was applied [61]. The method for the melody, bass line, and chord transcription used the same estimator, however, with an important difference: the estimator was used only to produce the pitch salience function without F0 detection. This way the decision of sounding pitches is postponed to the statistical framework. In addition, calculating only the pitch salience function is computationally very efficient, since the auditory model and the iterative pitch detection and cancellation scheme are not needed.

The choice of the frame size naturally affects the time-frequency resolution whereas the frame hop determines the temporal resolution of the transcription. Both of these also affect the computational complexity. For complex polyphonic music, the frame size of 92.2 ms is a good choice to capture F0s also from the lower pitch range (e.g., in the bass line transcription). The frame hop of 23.2 ms provides a reasonable temporal resolution, although for very rapid note passages, a smaller

frame hop (together with a smaller frame size) should be considered. The number of F0s in the output directly affects the computational complexity whereas the considered F0 region is selected according to the transcription target.

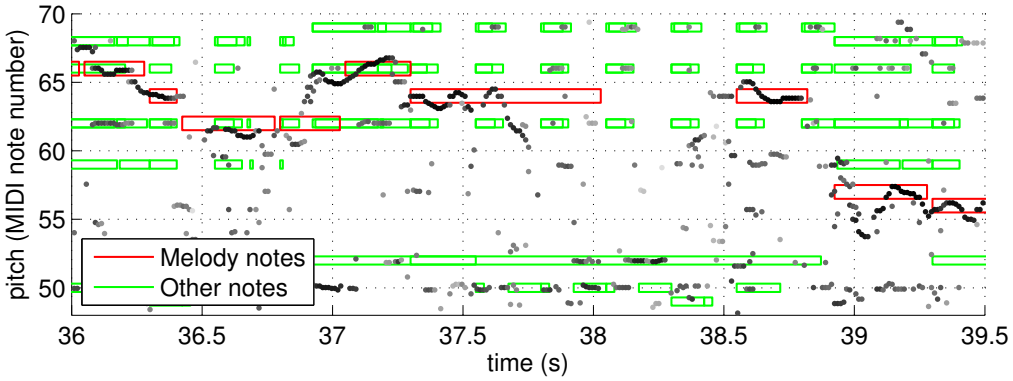
The frame-to-frame time difference of the salience values can also be calculated as in [P4]. The differential salience values are important for singing melody transcription since they indicate regions of varying pitch, e.g., in the case of glissandi and vibrato (see [108] for a discussion on singing sounds). In [P1] and [P2], only the positive changes were calculated and exposed to periodicity analysis for indicating onsets of pitched sounds. The temporal variation of pitch was found to be a useful cue for singing voice detection also in [32].

Figures 3.2 and 3.3 illustrate the outputs of the multiple-F0 estimation using different configurations for the middle and low pitch regions, respectively. The input signal is a short excerpt of the song RWC-MDB-P-2001 No. 6 from the Real World Computing (RWC) Popular music database [38]. The database includes manually prepared annotations of the sounding notes in MIDI format and they are shown in the figures by colored rectangles.

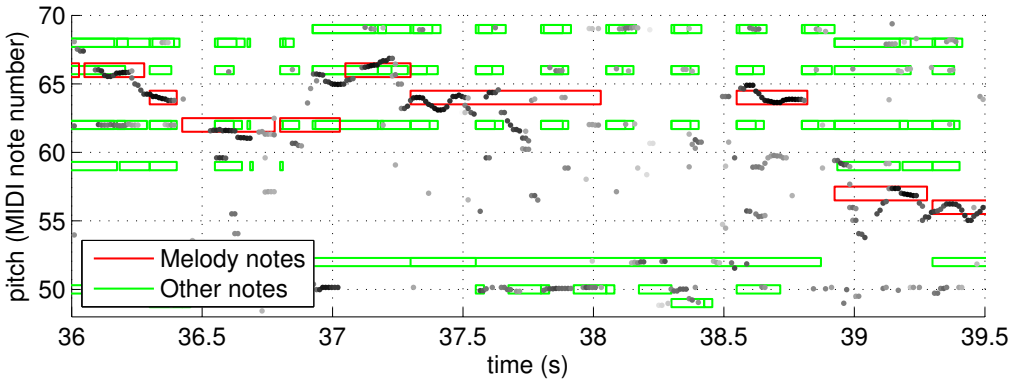
3.2 Accent Signal and Meter Analysis

Along with pitch features, the transcription methods employ features to facilitate note segmentation. *Accent signal* measures the amount of incoming spectral energy in time frame t and is useful for detecting note onsets. Calculation of the accent feature has been explained in detail in [64]. Briefly, a "perceptual spectrum" is first calculated in an analysis frame by measuring log-power levels within critical bands. Then the perceptual spectrum in the previous frame is element-wise subtracted from the current frame, and the resulting positive level differences are summed across bands. This results in the accent signal which is a perceptually-motivated measure of the amount of incoming spectral energy in each frame.

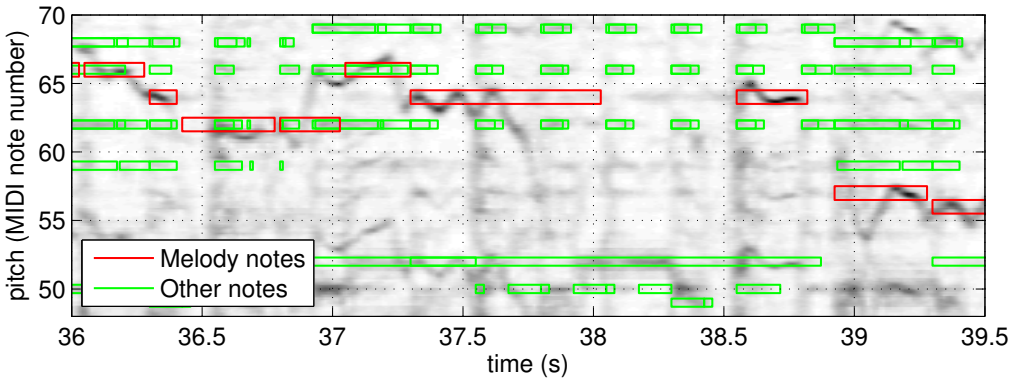
In addition, a more elaborate tempo and meter analysis was used to derive a metrical accent function in our work on monophonic singing transcription in [109]. This feature predicts potential note onsets at the metrically strong positions (e.g., at the estimated beat times) even when the audio signal itself exhibits no note onsets, e.g., in the accent signal. However, the advantage of using the metrical accent was found to be insignificant compared to the increased complexity of the tran-



(a) Auditory-model followed by iterative detection and cancellation.

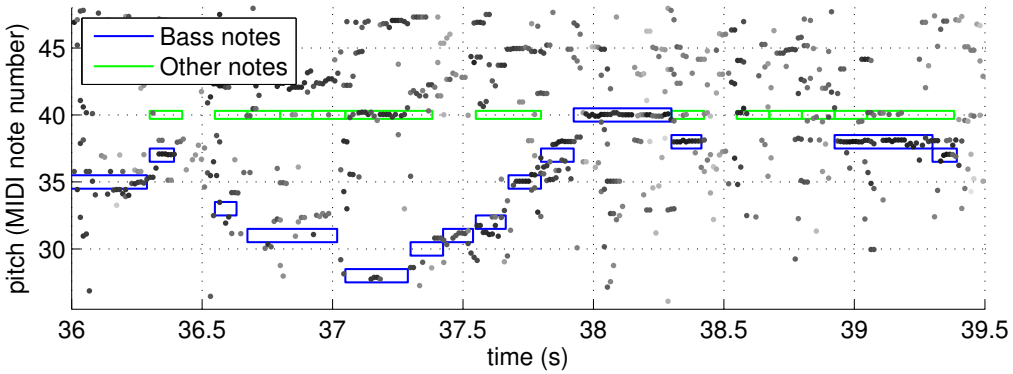


(b) Spectral whitening followed by iterative detection and cancellation.

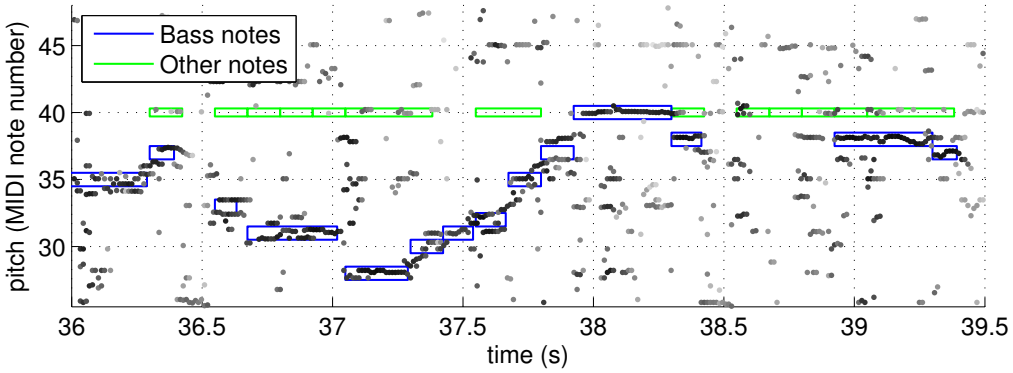


(c) Spectral whitening followed by pitch salience calculation.

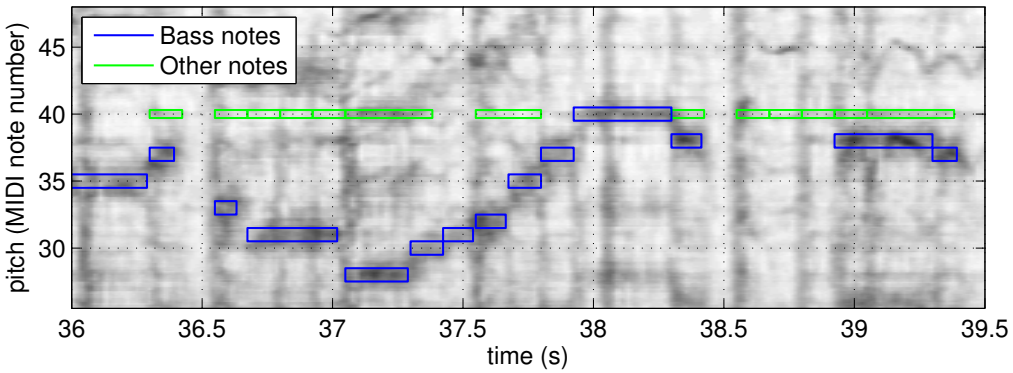
Figure 3.2: Features obtained by multiple-F0 estimation for an excerpt of RWC-MDB-P-2001 No. 6.



(a) Auditory-model followed by iterative detection and cancellation.



(b) Spectral whitening followed by iterative detection and cancellation.



(c) Spectral whitening followed by pitch salience calculation.

Figure 3.3: Features obtained by multiple-F0 estimation for the lower pitch region (RWC-MDB-P-2001 No. 6).

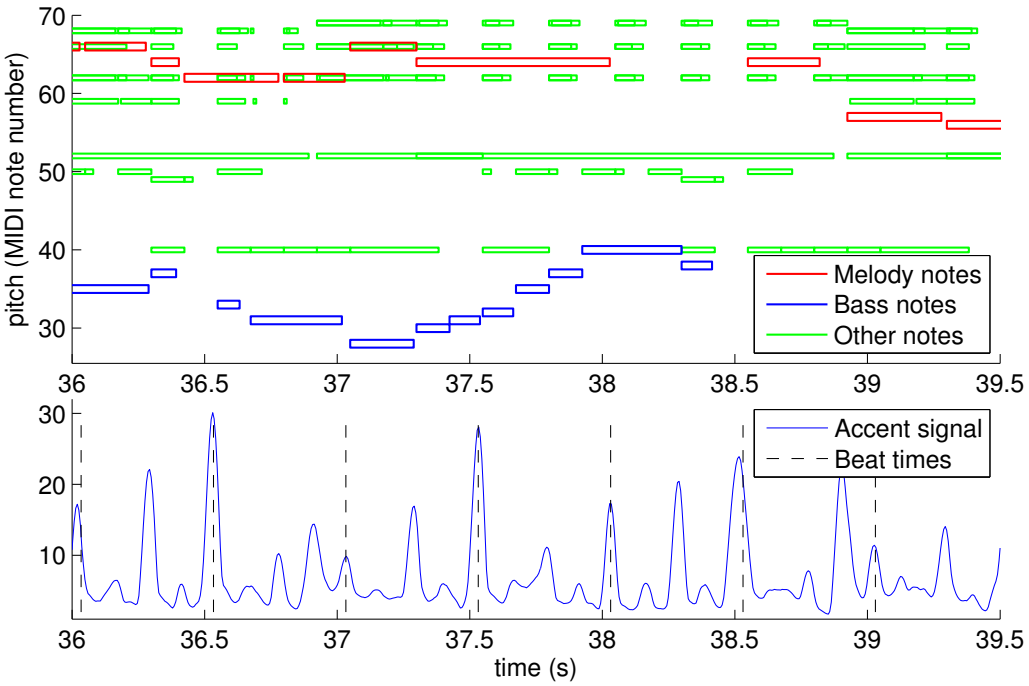


Figure 3.4: An example of the accent signal and the estimated beat times.

scription system in that case. For details on the feature and results, see [109]. Figure 3.4 shows an example of the accent signal and the estimated beat times for the same song excerpt that was shown in Figures 3.2 and 3.3.

3.3 Discussion

Although the proposed methods employ only the above-mentioned features, the framework allows the straightforward use of other features and feature extractors, too. Multipitch analysis could be carried out by using methods by Goto [35, 36], Dressler [25], Zhou [141], and Pertusa and Iñesta [97], to name a few examples. Seppänen *et al.* proposed a computationally streamlined beat-tracking method which also calculates the accent signal similar to the one used in this work [112]. The beat-tracking method by Dixon, called BeatRoot, could also be used [23].[2]

[2]Available for download at www.elec.qmul.ac.uk/people/simond/beatroot

Features accounting for timbre, such as *mel-frequency cepstral coefficients* (MFCCs) and their time derivatives, are commonly applied in instrument recognition and can be applied as features for the transcription task, too. For example, MFCCs have been used for detecting singing voice segments in polyphonic music [90, 59]. Alternatively, the strengths of harmonic partials for each F0 can be modeled to take timbre into account (as in the tone model by Goto, for example). Fujihara *et al.* estimated several F0 trajectories, used sinusoidal modeling to resynthesize the audio of each trajectory, and applied pre-trained models of vocal and non-vocal segments to evaluate the probability that a F0 candidate belongs to the vocal melody [32]. This was reported to improve the F0 estimation accuracy of vocal melodies. In [P2], we tested a rather similar idea but achieved no improvement in our preliminary simulations. Eggink and Brown employed an instrument recognition module, which used e.g. the partial strengths and their time differences as features, in melody transcription [28]. Li and Wang [71] applied the features of Eggink and Brown in an instrument recognition module which was used together with a transcription framework similar to ours.

In this work, all the features are extracted from single-channel audio signals, where possible stereo inputs are mixed to mono prior to the extraction.[3] However, spatial information in stereo recordings should be utilized as well and it has been successfully applied in sound-source separation methods (e.g., [3, 76, 128]). Parameters, which describe the left-right panning position of F0 estimates within stereo recordings, could be estimated and used as features.

[3] In [P1], multipitch estimation is performed for both left and right channels in the case of stereo input signals but the possible inter-channel dependencies are not utilized.

Chapter 4

Acoustic Modeling

Here the term acoustic modeling refers to learning statistical models for the values of extracted features during note events and rests. The parameters for the models are estimated from music recordings where the sounding notes have been annotated. The transcription methods use the models for calculating likelihoods for notes with different pitches and for rest segments in the music signal under analysis.

Hidden Markov model is an excellent tool for this task with well-established theory and algorithms. Briefly, HMM is a state machine consisting of a set of states $Q = \{q_1, \ldots, q_{|Q|}\}$, where $|Q|$ denotes the number of states. Let $r_t \in Q$ denote the random state variable of the machine at time t. A HMM can then be defined by the following parameters: state-transition probabilities $\mathsf{P}(r_t = q_j | r_{t-1} = q_i)$, the observation likelihood distribution $\mathsf{P}(\mathbf{o}_t | r_t = q_j)$, and the initial state probabilities $\mathsf{P}(r_1 = q_i)$. The actual state of the machine at time t cannot be directly observed (hence the name hidden) but it is estimated based on an observation vector $\mathbf{o}_t$ and its state-conditioned likelihood distribution.

Once we have the HMM parameters, the optimal state sequence $r^*_{1:T}$ explaining a sequence of observations $\mathbf{o}_{1:T} \equiv \mathbf{o}_1 \mathbf{o}_2 \ldots \mathbf{o}_T$ is given by

$$r^*_{1:T} = \underset{r_{1:T}}{\operatorname{argmax}} \left[\mathsf{P}(r_1)\mathsf{P}(\mathbf{o}_1|r_1) \prod_{t=2}^{T} \mathsf{P}(r_t|r_{t-1})\mathsf{P}(\mathbf{o}_t|r_t) \right], \tag{4.1}$$

which can be efficiently found by using the Viterbi algorithm [31], for example. A detailed introduction to HMMs is given in [101].

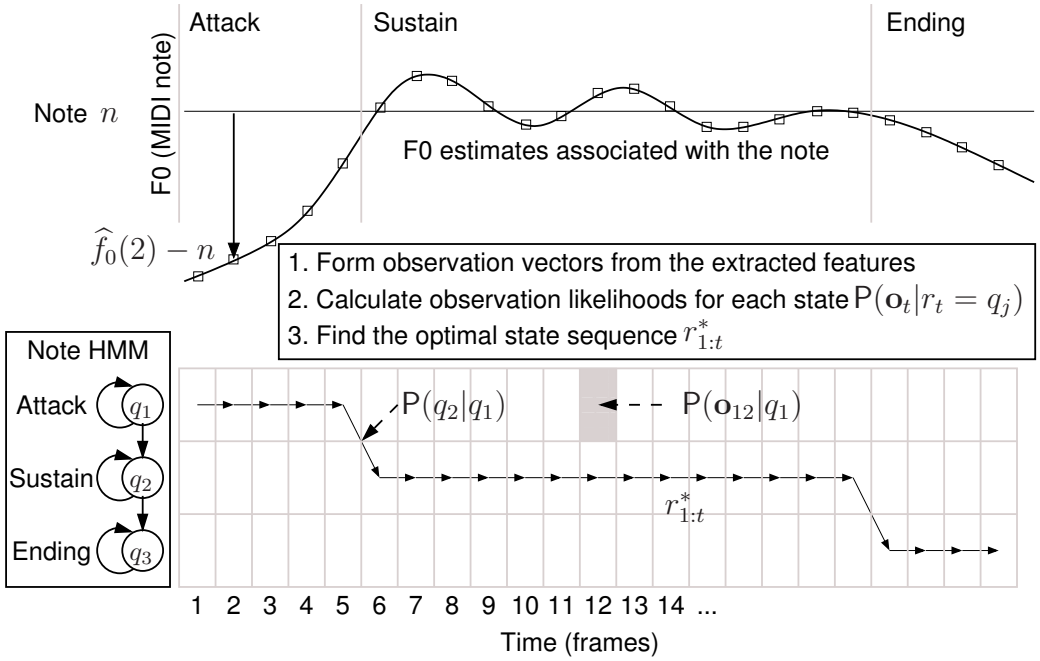


Figure 4.1: Note HMM accounting for a single note pitch n.

4.1 Note Event and Rest Modeling

The proposed methods use a three-state left-to-right HMM for note event modeling. State q_i represents the typical feature values in the i:th segment of a sounding note and the three states can be interpreted as the attack, sustain, and ending of a note event (see [108] for a discussion). Figure 4.1 illustrates the basic idea of the model accounting for a single note pitch n. To exemplify, let $\widehat{f}_0(t)$ denote the fundamental frequency estimate associated with the note, $s(t)$ its salience, and $a(t)$ the accent signal in frame t. The observation vector for the note is then defined by

$$\mathbf{o}_{n,t} = [\widehat{f}_0(t) - n,\ s(t),\ a(t)]^\mathsf{T}. \tag{4.2}$$

There are usually several F0 estimates around each note pitch n as illustrated in Figures 3.2 and 3.3. Therefore, the maximum-salience F0 estimate in the vicinity of the note pitch is often associated with the note (see the publications for details).

After this, the observation likelihoods $\mathsf{P}(\mathbf{o}_{n,t}|r_t = q_j)$ for each state q_j at each time t are calculated based on the HMM parameters. The optimal state sequence $r^*_{1:T}$ can then be resolved by Eq. 4.1, which is shown with the arrowed line in Figure 4.1. The change between the

model states in frames 5 and 6 indicates the temporal position of a transition from the attack state to the sustain state, for example.

As already introduced in Figure 4.1, the methods use the distance between a fundamental frequency estimate and the modeled note pitch n instead of the actual value of F0 estimate. This makes the note event model parameters independent of the actual note pitch and has the advantage that only one set of HMM parameters needs to be trained for the note event model in order to represent different note pitches. However, the observation vectors are specific to each note pitch n, as exemplified in Eq. 4.2. The form of the observation vector is specified for each transcription method in the included publications.

The parameters for the note event HMM are trained from audio recordings with manually labeled note events. For example, the RWC databases contain such annotations for a number of musical recordings [38, 39] and are very useful for the task (Figures 3.2 and 3.3 showed an example of the labeled note events). For the time region of each annotated note, the observation vectors constitute a training sequence for the model. The HMM parameters are then obtained using the Baum-Welch algorithm [101] where observation likelihood distributions are modeled with *Gaussian mixture models (GMMs)*.

In addition to note events, the methods use a model for rests, i.e., the segments where no notes are sounding. Rests are modeled with a one-state HMM which is here analogous to a GMM, since there is no hidden state if only one state is possible. Usually the maximum-salience value in frame t is used as a component of the observation vector $\mathrm{o}_{r,t}$ for rest, and the observation likelihood $\mathsf{P}(\mathrm{o}_{r,t})$ is calculated using the learned rest GMM parameters, as in [P2] and [P3]. In [P1], the likelihood for rest is derived from the note observation likelihoods, and in [P4], the rest state corresponds to using the "background" explanation for all notes (see [P4] for the details).

Using the Models

Now we have models for a note with pitch n and rest segments. In order to transcribe music, the note model structure shown in Figure 4.1 is replicated for each note pitch in the desired note range, e.g., for MIDI notes 44–84, and combined with the rest model. This results in a network of note models and the rest model as illustrated in Figure 4.2. Given the frame-wise extracted features, the observation likelihoods are calculated for each state of each note and the rest state based on the

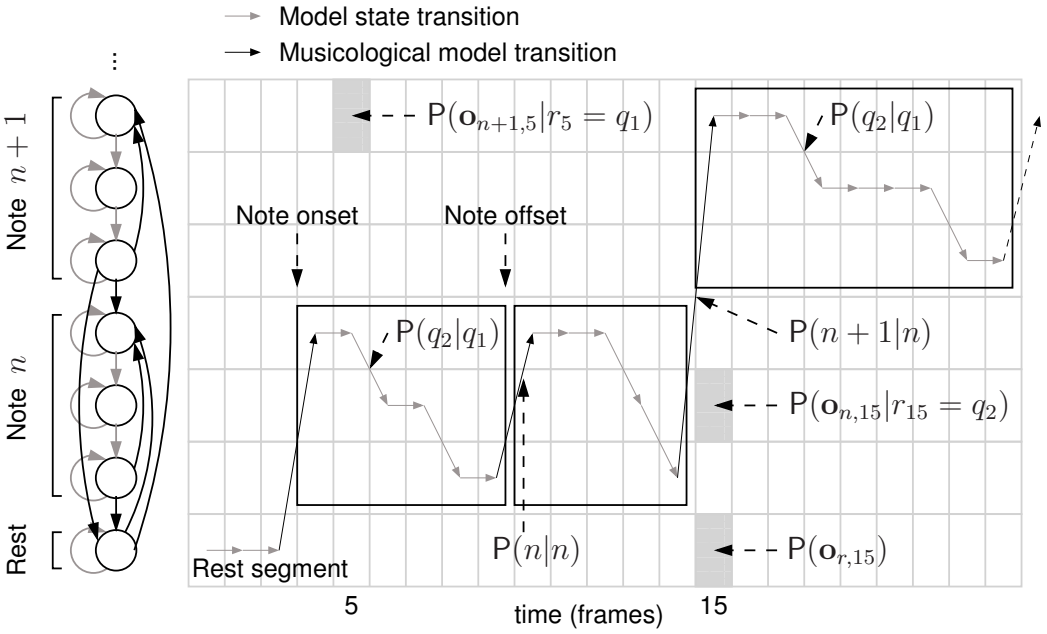


Figure 4.2: A network of note models and the rest model for music transcription.

model parameters. In addition to the state-transitions within the models, the methods need probabilities for the transitions between notes and rests, i.e., the musicological model. Obtaining these probabilities will be discussed in Chapter 5.

The transcription is obtained by finding a path through the network (indicated with the arrowed line in Figure 4.2) by solving Eq. 4.1. This simultaneously produces the discrete pitch labels and note segmentation, i.e., the note onsets and offsets. A note starts when the path enters the first state of a note model and ends when the path exits its last state. Rests are produced when the path goes through the rest state. Figure 4.2 shows three transcribed notes (with discrete note pitches $n, n, n+1$) and their onset and offset times. There is also a rest segment in the first few frames. In [P1], a polyphonic transcription output is obtained by transcribing several note sequences in an iterative manner. The method simply finds a path through the network, prohibits using the transcribed note segments on the next iteration, and repeats this as long as new notes are found or until the desired polyphony is reached.

The path can be found using the Viterbi algorithm as used in [P2], [P3], and [P4]. Another alternative, the token-passing algorithm [140],

was applied in [P1]. The token-passing algorithm was originally designed for large-vocabulary speech recognition and is very similar to the Viterbi algorithm. The Viterbi decoding relies on the first-order Markov assumption (i.e., a random state is conditionally dependent only of the preceding state) whereas the token-passing algorithm can store several best paths (or, tokens) in each frame and possibly find a better global path through the network if the first-order Markov assumption does not hold. This happens when the musicological model uses several previous notes to determine the transition probability, or a key estimate is updated in a causal manner. In practise, however, there is no significant difference in the transcription results between the decoding algorithms, and thus we prefer using the Viterbi decoding due to its simple implementation and very straightforward processing. The above discussion is affirmed by the results for the bass line transcription from streaming audio [P3]; the difference in the transcription results was negligible between the standard Viterbi decoding (the first-order Markov assumption held) and a suboptimal Viterbi with a block-wise backtracking and a causally updated key estimate. See [P3] for details.

4.2 Contrasting Target Notes with Other Instrument Notes and Noise

The transcription method for the melody, bass line, and chords [P4] uses three types of note event models instead of a single model for the target notes. The basic idea is that all the considered note pitches at all times are classified as target notes (melody or bass), as notes from the other instruments, or as noise or silence. The use of the target-notes and the other-notes models aims at improving the discriminability of the target sound source from other instruments. Details are given in [P4].

Chapter 5

Musicological Modeling

Chapter 4 introduced the acoustic modeling of individual note events and rests without using knowledge on the other simultaneously or previously sounding notes in the music piece. The musical context, however, plays an important role in how notes are arranged and related to the harmony of the piece. In other words, some notes and note sequences are more probable than others when considering the musical context. As an example, a note sequence C, E, G (the notes of C major chord) in the key of C major is musically very natural. Shifting the last note of the sequence, G, up by a semitone to G♯ results in the notes of C augmented chord. If the individual note models for pitches G and G♯ give approximately equal likelihoods for the notes, the methods can utilize musical context (e.g., the key of C major and the previous notes) and prefer transcribing the more common sequence which ends in note G.

The proposed framework for music transcription enables utilizing musicological knowledge in the transcription in a straightforward manner by assigning probabilities for the transitions between notes and rests. The proposed methods use this feature by first estimating the musical key of the piece and then using key-dependent transition probabilities, trained with note sequences from MIDI files. The following sections briefly introduce the key estimation and the training of the note-transition models, with a discussion about chord transcription. The details of each method are given in the enclosed publications.

5.1 Key Estimation

Several methods have been proposed for the key estimation and the analysis of chord progression from music signals, including [100, 34, 95, 87, 52, 67]. In order to analyze simultaneously sounding notes, pitch saliences are commonly mapped to a *pitch-class representation*. Briefly, the set of notes which belong to a pitch class $m \in \{0, 1, \ldots, 11\}$ is defined by $\mathcal{H}_m = \{n \mid n \in \mathcal{N} \wedge \text{mod}(n, 12) = m\}$, where $\mathcal{N}$ is the note range and $\text{mod}(x, y) \equiv x - y\lfloor x/y \rfloor$. The pitch-class profile $\text{PCP}_t(m)$ measures the salience of pitch class m in frame t, for example by summing up the saliences of notes belonging to the pitch class. This type of representation is also referred to as the *chroma* vector. The calculation of the pitch-class profile varies between different methods but all of them bear information on how spectral energy is distributed among the pitch classes. This representation is extensively used in several harmonic analysis methods, including the chord transcription method in [P4]. The pitch-class representation can be further mapped to a musically more relevant representation, such as the tonal centroid [46]. The tonal centroid is a feature vector based on the harmonic network, or *Tonnetz* (see [100, 34]), and the idea is that two tonal-centroid vectors mapped from pitch classes with close harmonic relations (e.g., fifths, major and minor thirds) are close to each other in the Euclidean space. The tonal centroid has been applied in the detection of harmonic changes [46] and in key and chord transcription [67], for example.

Commonly, the pitch-class profile, or the like, is used as an observation vector $\mathbf{o}_t$ for a HMM with states representing chords, keys, or chord transitions, as in [113, 5, 87, 93, 67]. The model parameters are obtained either by training them from audio with labeled chord and key segments, or then using reported pitch-class distributions for different keys or chords. The latter include, for example, the pitch-class distributions reported by Krumhansl [65] shown in Figure 5.1. Once the parameters have been obtained, the model can assign a likelihood for each key or chord given the observation in a frame. Let this be denoted by $\mathsf{P}(\mathbf{o}_t | r_t = q_j)$ where the state q_j can be one among i) a set of chords, $j \in 0, \ldots, 23$, to represent twelve major and twelve minor triads or ii) a set of relative-key pairs, $j \in 0, \ldots, 11$, to represent pairs [C major / A minor], [D♭ major / B♭ minor], and so forth until the pair [B major / G♯ minor]. If key or chord transitions are defined as well, $\mathsf{P}(r_t = q_j | r_{t-1} = q_i)$, the Viterbi algorithm can be used to decode a sequence of keys or chords using Eq. 4.1.

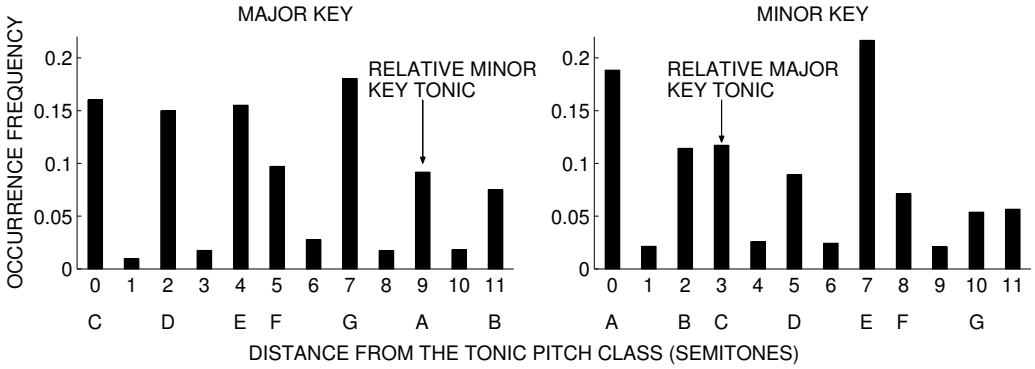


Figure 5.1: Pitch class occurrence frequencies in major and minor keys with respect to the tonic pitch class reported by Krumhansl [65, p. 67]. As an example, the pitch class names are listed below the figure axes for the relative keys C major and A minor. (After [108, p. 386].)

The proposed methods use a key estimation method to decide the relative-key pair of the music piece under analysis. In [P4], for example, the key estimation first maps the values of the pitch salience function into a pitch-class profile. Then, the likelihood of a key is obtained by rotating the profile so that pitch class $m = 0$ corresponds to the tonic note of the key and comparing the rotated profile with the distributions reported by Krumhansl. The likelihoods are accumulated for each key over time, and the most probable relative-key pair is simply decided by the maximum likelihood among the keys. This corresponds to a key HMM with twelve states where, however, there exist no transitions between the states, i.e., $\mathsf{P}(r_t = q_j | r_{t-1} = q_i) \neq 0$ only if $i = j$. The chord transcription is similarly obtained, however, with chord transitions and profiles for major and minor chords estimated from manually annotated music.

Key estimation forms a basis for utilizing musical context, and naturally, for using key-dependent note-transition models. The key estimation method itself is not important as long as it produces somewhat correct estimates, and the proposed methods could apply any of the above-listed key estimation methods. Publication [P2] reports the influence of the perfect and the worst-case key estimates on the melody transcription results.

5.2 Note Sequence Modeling

The objective of note sequence modeling is to solve ambiguous note labelings by utilizing previous note pitches and the estimated key. Given a sequence of note pitches, $n_{1:t-1}$, the note sequence models give probabilities $\mathsf{P}(n_t|n_{1:t-1})$ for the following note n_t. It is not reasonable to model all the possible past note sequences, and therefore, the methods approximate the probabilities based on the most recent notes. Note N-gram models the probability of the note pitch based on $N-1$ previous note pitches:

$$\mathsf{P}(n_t|n_{1:t-1}) \approx \mathsf{P}(n_t|n_{t-N+1:t-1}). \tag{5.1}$$

Most of the methods apply *note bigrams* where $N = 2$. This reduces the model in Eq. 5.1 to $\mathsf{P}(n_t|n_{t-1})$ fulfilling the first-order Markov assumption. The models take into account only the pitch of the notes and not the temporal aspects such as onsets or durations.

The note bigram models are based on histograms obtained by counting note-interval occurrences from a collection of MIDI files where key information has been annotated. The Essen Associative Code and Folksong database (EsAC) is suitable for melody note sequences, including thousands of folksongs with key information.[1] Let $k_{\text{maj}} = 0, \ldots, 11$ and $k_{\text{min}} = 0, \ldots, 11$ denote the major and minor mode keys with tonic notes C, C♯, D, and so forth. A transition from note n to note n' is now defined by i) the degree of the first note n and ii) the interval $n' - n$ between the notes. The term *note degree* refers here to distance $\text{mod}(n-k, 12)$. Then these transitions are counted from the MIDI files.

The upper panels in Figure 5.2 show the obtained histograms of note transitions for major and minor keys in the EsAC database. The histograms show very clearly that most of the transitions occur within the diatonic scale. The example given at the beginning of this chapter can now be confirmed. For C major key, $k_{\text{maj}} = 0$, the transition from note E ($n = 52$) to G ($n' = 55$) results in note degree $\text{mod}(52 - 0, 12) = 4$ with interval $55 - 52 = 3$ which has occurred much more frequently than the transition to G♯ with interval 4.

Since key estimators easily confuse the major and minor mode, the major and minor histograms are combined to obtain a single distribution for relative-key pairs $k = 0, \ldots, 11$ corresponding to [C major / A minor], [D♭ major / B♭ minor], and so forth until the pair [B major / G♯ minor]. This is easily obtained by summing up the histograms and the result is shown in the bottom panel of Figure 5.2. However, we need to

[1] www.esac-data.org

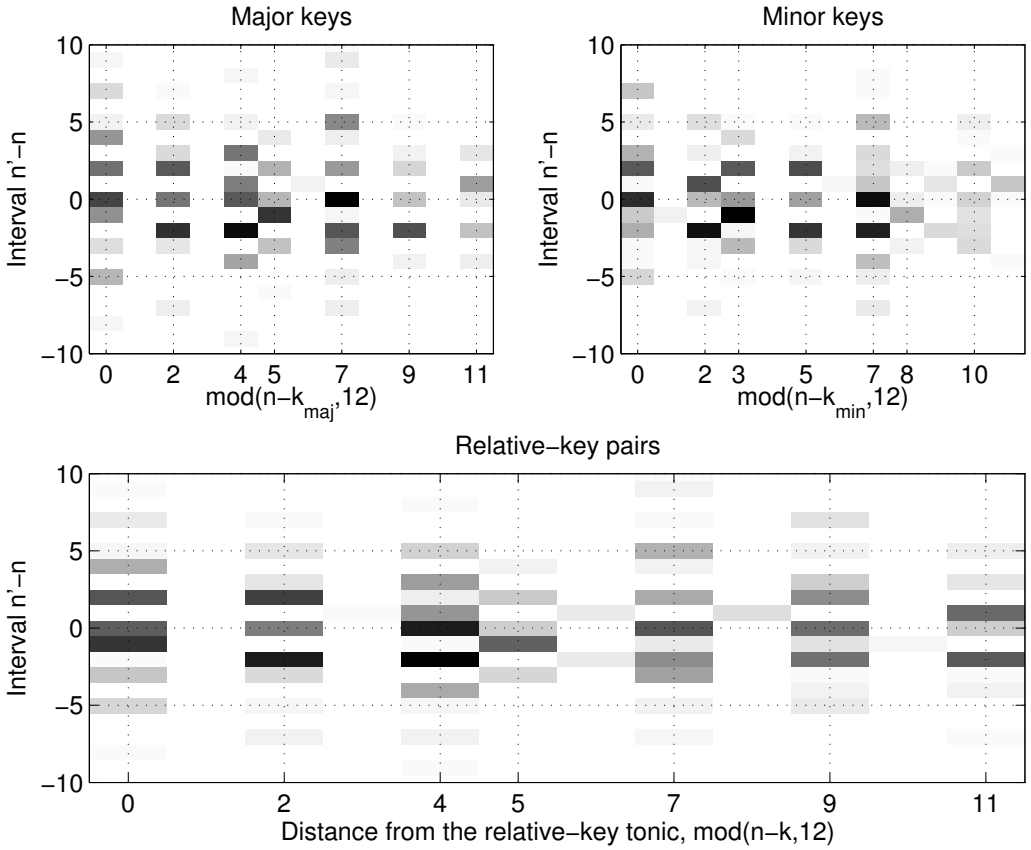


Figure 5.2: Note transitions in EsAC database.

rotate the histogram for minor keys so that the notes of A minor match the notes of C major, for example. In general, the relation between the relative major and minor keys obeys $k_{\text{maj}} = \text{mod}(k_{\text{min}} + 3, 12)$ and $k_{\text{min}} = \text{mod}(k_{\text{maj}} + 9, 12)$. For example, B minor $k_{\text{min}} = 11$ has relative D major key by $k_{\text{maj}} = \text{mod}(11 + 3, 12) = 2$.

The note transition histograms can now be used to build a key-dependent note bigram, which gives probabilities for transitions between notes once the relative-key pair k has been estimated. Figure 5.3 shows two example note bigrams for relative-key pairs [C major / A minor] and [F major / D minor] for note range C3–B3 (i.e., MIDI notes 48–59). The Witten-Bell discounting algorithm [135] is used for normalizing and smoothing the transition probabilities. In general, the transition probabilities over all states given the preceding state must sum to unity.

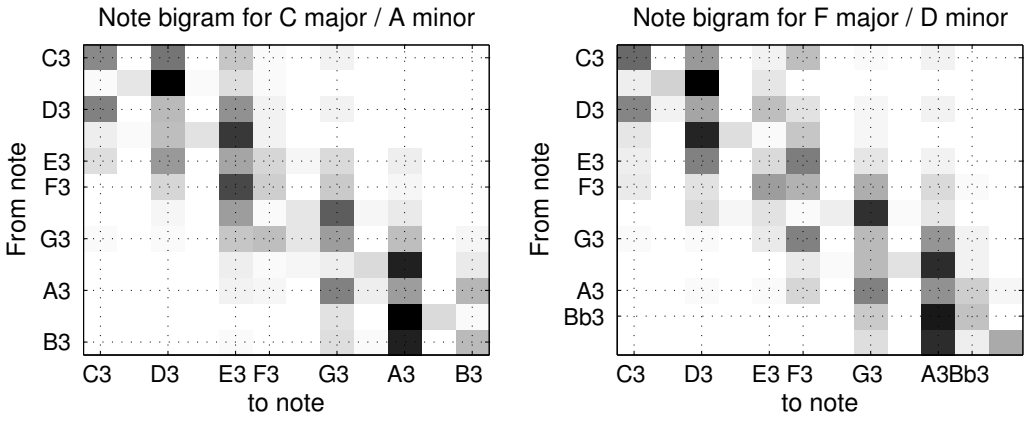


Figure 5.3: Two note bigrams for relative-key pairs [C major / A minor] and [F major / D minor] where the transition probability is indicated with gray-level intensity.

The note bigrams do not take into account the absolute pitch of the notes but only the interval between them. However, it is advantageous to prefer target notes in the typical pitch range of the target instrument. In [P4], this was implemented by weighting the probabilities with a normal distribution over pitches (see [P4] for details). The methods also take into account the transitions between notes and rests. The note-to-rest and rest-to-note transitions can be counted similarly to note transitions within the training material. Another alternative is to use, for example, the Krumhansl distributions. This assumes that note sequences tend to begin and end with the pitches on the diatonic scale.

N-gram modeling consists of $|Q|^N$ possible transitions, where $|Q|$ is the number of states. As a result, using the models with large values of N requires lots of training material and memory. Variable-order Markov model offers a useful alternative for note pitch prediction where the context length varies in response to the available statistics in the training data [4]. This is a very desirable feature, and for note sequences, this means that a single model can take into account both short and long note sequences based on their occurrence in the training data.

VMM prediction was applied instead of N-grams to capture the repetitive nature of bass note sequences [P3]. The VMM was trained using bass note sequences in a collection of MIDI files. In addition, it is likely that the same note sequences, possibly with slight variations,

are repeated several times within a song. We took this into account by an optional post-processing step in [P3], where a song-specific VMM was trained with the first transcription of the bass line and used on the second pass of the Viterbi decoding. This slightly reduced transcription errors in our simulations.

5.3 Chord Sequence Modeling

In addition to note sequences, it is beneficial to utilize musicological modeling in transitions between chords, since there are common chord progressions in Western music. The chord transition model can be either estimated from data as in [P4], or then the model can be derived from music theory basics. As an example of this, Bello and Pickens used the distance on the circle of fifths for approximating probabilities for chord transitions [5].

Chapter 6

Transcription Methods and Evaluation

This chapter briefly introduces the evaluation criteria, used databases, and results for the proposed transcription methods. Some references are given to work by other researchers, where the transcription methods have been used in comparative evaluations. The chapter is concluded with transcription examples.

6.1 Evaluation Criteria and Data

For the note-based evaluation of the transcriptions, we used the recall rate R and the precision rate P defined by

$$R = \frac{\#(\text{correctly transc. notes})}{\#(\text{reference notes})}, \; P = \frac{\#(\text{correctly transc. notes})}{\#(\text{transcribed notes})}, \quad (6.1)$$

in [P1]–[P4]. A reference note was correctly transcribed by a note in the transcription if their discrete MIDI note numbers were equal, the absolute difference between their onset times was less than 150 ms, and the transcribed note was not already associated with another reference note. The F-measure $F = 2RP/(R + P)$ was used to give an overall measure of performance. Temporal overlap ratio of a correctly transcribed note with the associated reference note was measured by $\rho = (\min\{\mathcal{E}\} - \max\{\mathcal{B}\})/(\max\{\mathcal{E}\} - \min\{\mathcal{B}\})$, where sets $\mathcal{B}$ and $\mathcal{E}$ contained the beginning and ending times of the two notes, respectively. Mean overlap ratio $\bar{\rho}$ was obtained by averaging ρ values over the correctly transcribed notes.

For the sake of comparison with the melody transcription method by Ellis and Poliner [29], the method in [P4] was evaluated also with frame-based evaluation criteria. The criteria were adopted from the "audio melody extraction" task in MIREX 2005–2006. See [98] for details.

The chord transcription in [P4] was evaluated by comparing the transcribed chords with the reference chords frame-by-frame, including an error analysis and a comparison with the chord transcription method by Bello and Pickens [5].

All the proposed methods have been evaluated with RWC popular music (RWC-MDB-P-2001) and genre (RWC-MDB-G-2001) databases of polyphonic music [38, 39]. The databases contain a MIDI file for each song with manually annotated reference notes for the melody, bass, and other instruments. MIDI notes for drums, percussive instruments, and sound effects were excluded from the evaluations. The results were obtained via cross validation since the databases have been also used for training the acoustic models. Notice that some of the database songs were omitted due to unreliable synchronization with the reference notes, or due to the absence of the transcription target (e.g., the melody or the bass line).

For chord transcription in [P4], we used the first eight albums by The Beatles with annotations provided by Harte *et al.* [45]. This database was used both for the training and the evaluation with cross validation.

6.2 Results

Table 6.1 summarizes the reported results and the used data for the proposed transcription methods. Detailed results for different method configurations are given in the publications. To summarize, recall rate of 60% was achieved for the melody and bass line with precision varying around 50% for polyphonic popular music. For varying styles of music in RWC genre database, the performance was somewhat lower than for popular music. Transcription of all the pitched notes achieved the lowest results where, however, the task was also the most challenging. In the evaluation of the method [P1], it was required that also the reference notes with colliding pitch and timing (about 20% of the notes) are transcribed although the method is not capable of transcribing such notes. The chord transcription gave correct frame-based labeling for about 70% of the time.

Table 6.1: Summary of the note-based evaluation results for the transcription methods.

Publication	Target	Database(s)	Songs	Clips	Total amount of audio (min)	R	P	F	$\bar{\rho}$	Speed
[P1]	All pitched notes	RWC-MDB-G-2001	91[a]	30 s	45.5	39	41	40	40	0.3
[P2]	Singing melody	RWC-MDB-P-2001	96[b]	60 s	96	63	45	51	53	1.0
[P3]	Bass line	RWC-MDB-P-2001	87[c]	60 s	87	63	59	59	62	7.3
[P4]	Melody	RWC-MDB-P-2001+ RWC-MDB-G-2001	92+38	full	524	55	50	51	60	11.4
		RWC-MDB-P-2001	92[d]		375	61	49	54	61	
		RWC-MDB-G-2001	38[e]		149	42	50	43	56	
[P4]	Bass line	RWC-MDB-P-2001+ RWC-MDB-G-2001	84+43	full	509	50	58	51	60	11.4
		RWC-MDB-P-2001	84[f]		342	58	58	56	62	
		RWC-MDB-G-2001	43[g]		168	35	58	39	58	
[P4]	Chords	First eight Beatles albums	110	full	276	71% of frames correct				14.1

[a] RWC-MDB-G-2001 Nos. 50, 52–53, 56–57, 80, 88–89, and 97–99 omitted (No. 58 has three parts evaluated separately).
[b] RWC-MDB-P-2001 Nos. 27, 31, 43, and 74 omitted.
[c] RWC-MDB-P-2001 Nos. 10, 33, 71–80, and 99 omitted.
[d] RWC-MDB-P-2001 Nos. 10, 33–34, 38, 56, and 71–73 omitted.
[e] RWC-MDB-G-2001 Nos. 1–6, 8–9, 11–13, 16–18, 20, 22, 24–27, 35, 39–41, 45–46, 65–69, 73, 75, 78, 91–93, and 100 used.
[f] RWC-MDB-P-2001 Nos. 10, 33–34, 38, 56, 71–80, and 99 omitted.
[g] RWC-MDB-G-2001 Nos. 1–6, 8–9, 11–22, 24–27, 35–36, 39–46, 65–67, 69, 73, 75, and 91–93 used.

The table also lists the execution speed of the methods for monaural audio input compared with real-time processing (i.e., the duration of input signal is divided by the execution time). These values were obtained with simulations on a 3.2 GHz Pentium 4 processor. The latest transcription methods, e.g., [P3] and [P4], are the most efficient in terms of the execution time.

6.3 Comparative Evaluations

The proposed transcription methods have achieved good results in comparative evaluations with other state-of-the-art methods. For example, the transcription method [P1] was submitted to task "Multiple Fundamental Frequency Estimation & Tracking" in MIREX 2007 and 2008 evaluations [1, 2]. The task included two subtasks: multi-F0 estimation and polyphonic note tracking. In the multi-F0 estimation subtask, the aim was to frame-wise extract multiple pitches from polyphonic music recordings. The second subtask, polyphonic note tracking, required transcribing pitched note events in music recordings.

Figure 6.1 summarizes the results for the subtasks in MIREX 2007 evaluation. The proposed method (team "RK") was ranked first in both of them (see [1] for detailed results and method abstracts). In the frame-based multiple-F0 estimation subtask, the differences in the results were negligible between the first four methods. The evaluation criteria for this subtask are defined in [99]. Since the proposed method was originally designed to produce MIDI notes as an output, the differences were more pronounced in the subtask of polyphonic note tracking (the bottom panel in Figure 6.1). The method also performed best for piano material although the note model parameters were trained to obtain a generic music transcription method without specializing for any particular instrument. The trained note-model parameters were the same ones used in the simulations of the original publication [P1].

Figure 6.2 shows the corresponding results in MIREX 2008 evaluation [2], where our results were identical to the ones in 2007 evaluation. Pleasingly, the results were improved by other teams, including the submissions of Yeh *et al.* (team "YRC") and Pertusa & Iñesta (team "PI") in the frame-based evaluation. Our submission was ranked as fourth in this subtask. In the note-tracking subtask (the bottom panel in Figure 6.2), our method [P1] still outperformed other submissions according to the "onset only" criterion (as in the 2007 evaluation). However, if the note offsets were taken into account in the evaluation cri-

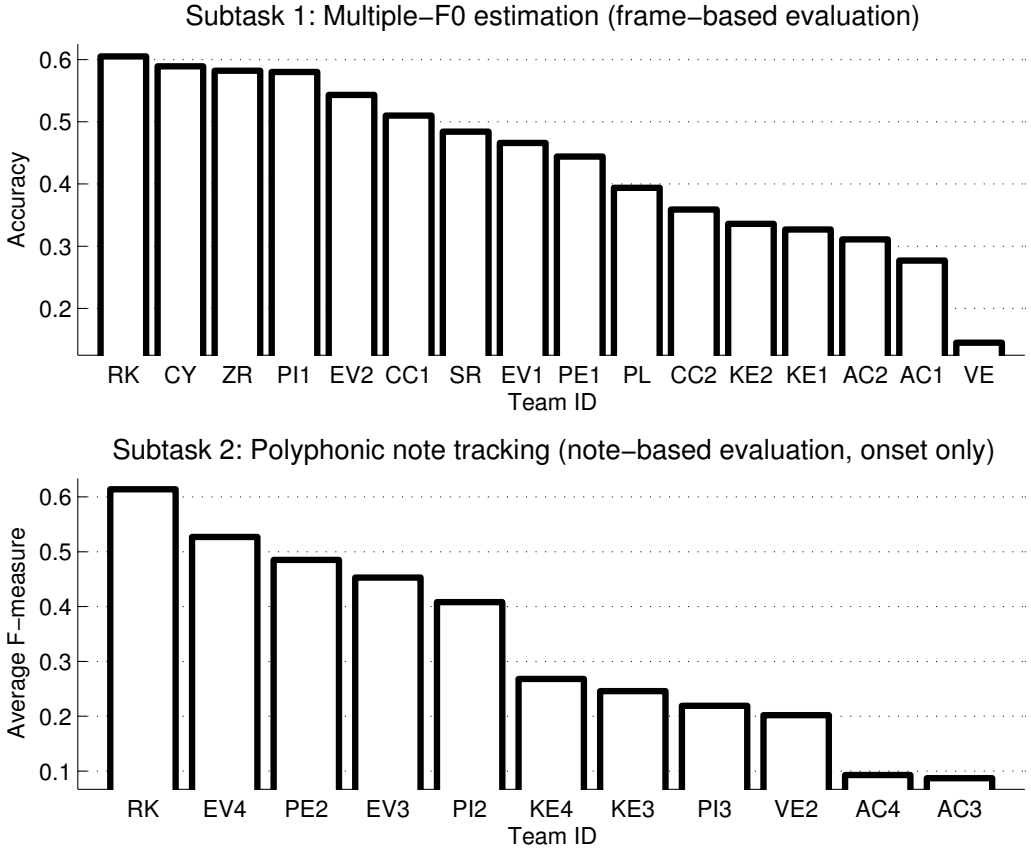


Figure 6.1: MIREX 2007 results for "Multiple Fundamental Frequency Estimation & Tracking" task. See text for details.

terion, the submission by Yeh *et al.* [137, 11] performed slightly better (see [2] for the detailed results).

The good general performance of the method was also noted by Poliner and Ellis in [99]. They compared the results of the proposed method [P1], the transcription method by Marolt [74], and their transcription method in a polyphonic piano transcription task. See [99] for details.

The proposed methods for melody transcription [P2] and [P4] have also been evaluated in "Audio Melody Extraction" task in MIREX evaluations 2005, 2006, and 2008. The goal was to estimate F0 trajectory of the main melody within polyphonic music. In 2005, we submitted a version of the polyphonic transcription method [P1] which was then developed into the melody transcription method [P2] for the 2006 sub-

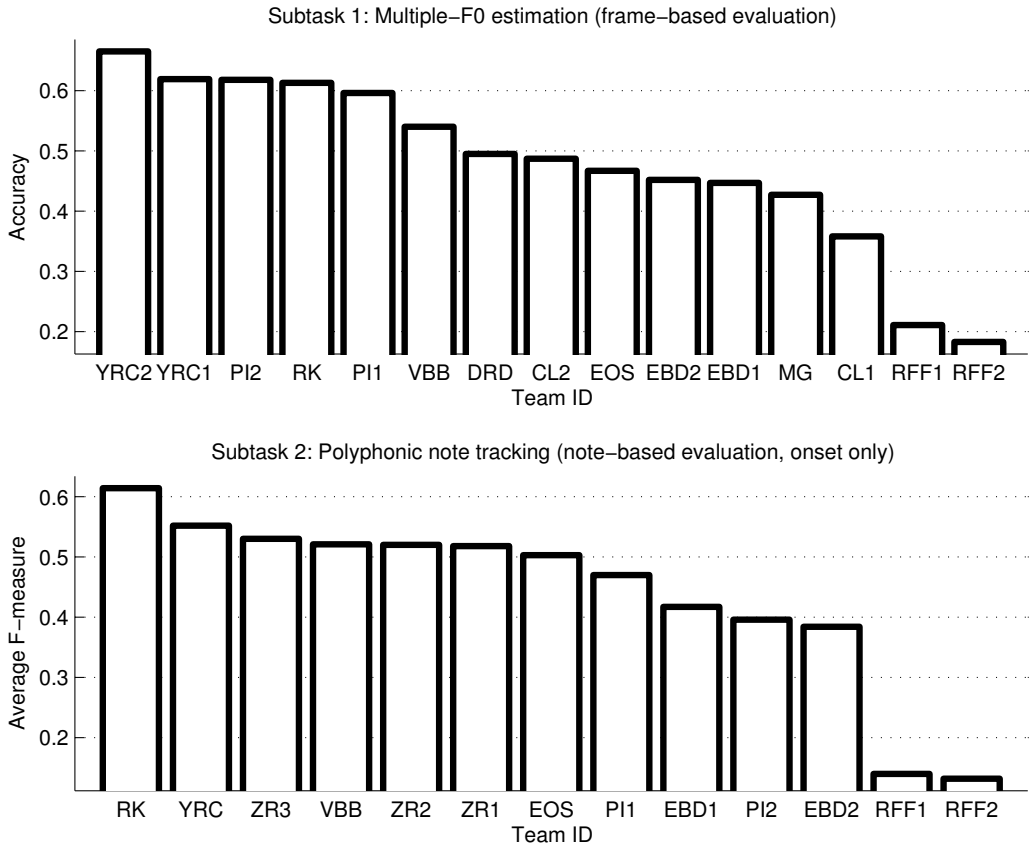


Figure 6.2: MIREX 2008 results for "Multiple Fundamental Frequency Estimation & Tracking" task. See text for details.

mission[1]. The proposed methods were ranked second best both in 2005 and in 2006. The different approaches for the 2005 evaluation are compared in detail in [98]. In 2008, the melody transcription method [P4] was first used for producing a note-level transcription. This was followed by the estimation of a detailed F0 trajectory for each note as described in [P6]. The submitted method was ranked as third with overall accuracy of 71.1% [2]. The methods by Cancela and by Durrie *et al.* performed clearly better with overall accuracies of 76.1% and 75.0% whereas our method ran over 400 and 20 times faster than these methods, respectively. Comparative results of the melody transcription method [P2] have been also reported by Li and Wang in [72].

[1]For the detailed results of the 2006 evaluation, see www.music-ir.org/mirex2006/index.php/Audio_Melody_Extraction_Results.

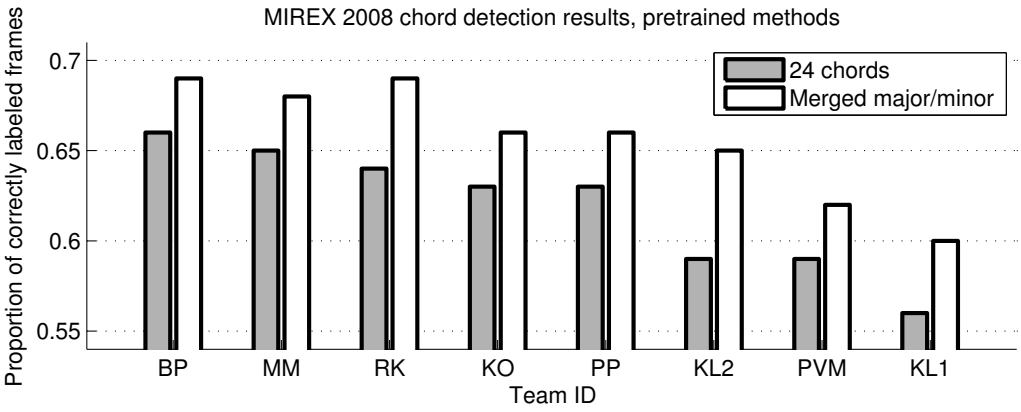


Figure 6.3: MIREX 2008 results for "Audio Chord Detection" task.

The chord transcription method [P4] was evaluated in MIREX 2008 "Audio Chord Detection" task [2]. The evaluation dataset consisted of 176 full songs by The Beatles where the annotations provided by Harte *et al.* [45] were used as the ground-truth. The submitted methods were required to transcribe major/minor triads and no-chord segments. Evaluation was performed using a frame-based criterion similar to the one used in [P4]. In addition, the evaluation was arranged for both pre-trained methods (the parameters of the methods were fixed) and methods which were run in three-fold cross validation. Our method was evaluated in the former category.

Figure 6.3 shows the results for the pre-trained methods. The best performing method by Bello & Pickens (team "BP") labeled correctly 66% of the frames. Due to the rather similar approaches (namely using HMMs) to the task, it is not surprising that differences in the results are very small, e.g., Mehnert 65% (team "MM"), Ryynänen & Klapuri 64% (team "RK"), Khadkevich & Omologo 63% (team "KO"), and Papadopoulos & Peeters 63% (team "PP"). As noted in [P4] in the comparison with the method by Bello & Pickens, our method tends to confuse major and minor modes more often. When these errors are ignored (the white bars in Figure 6.3), our method performed as well as the method by Bello & Pickens (both 69% correct). Our method took about half an hour to transcribe the 176 songs and was the second fastest submission after the method by Lee (team "KL").

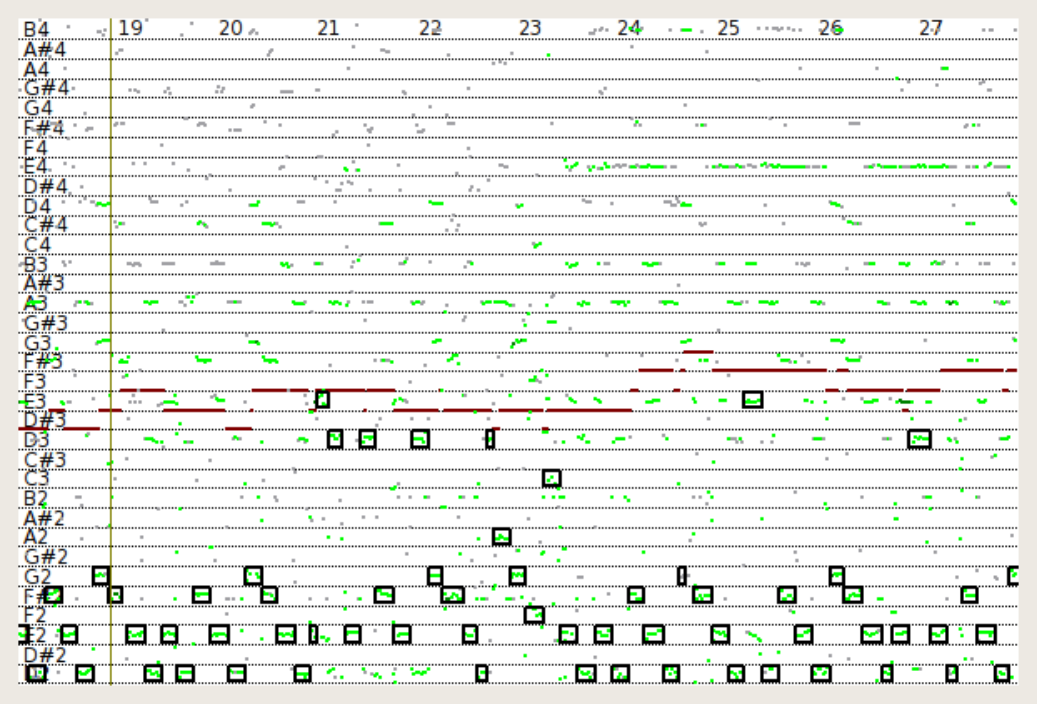


Figure 6.4: Bass line transcription from streaming polyphonic audio. See text for details.

6.4 Transcription Examples

This section shows a few transcription examples and graphical user interfaces (GUIs) for the proposed methods. Figure 6.4 shows an example of bass line transcription from streaming audio [P3]. The proposed method was implemented in C++ with a GUI. The implementation takes an audio file as the input and starts the audio playback and the transcription process simultaneously. Since here the method analyzes an audio file instead of an audio stream, the causal processing proceeds faster than the playback and the transcription results can be shown in advance. The vertical line shows the audio playback position, the black rectangles the transcribed bass notes, the green and gray dots the frame-wise extracted F0s, and the red line the estimated upper F0 limit for the transcription (see [P3] for details). The analyzed song in the example is “Ignotus Per Ignotum” by Planet X.

Figure 6.5 shows an example of the melody, bass line, and chord transcription [P4] of the song “Like a Virgin” by Madonna. The upper panel shows the transcription on a piano roll, where the red rectan-

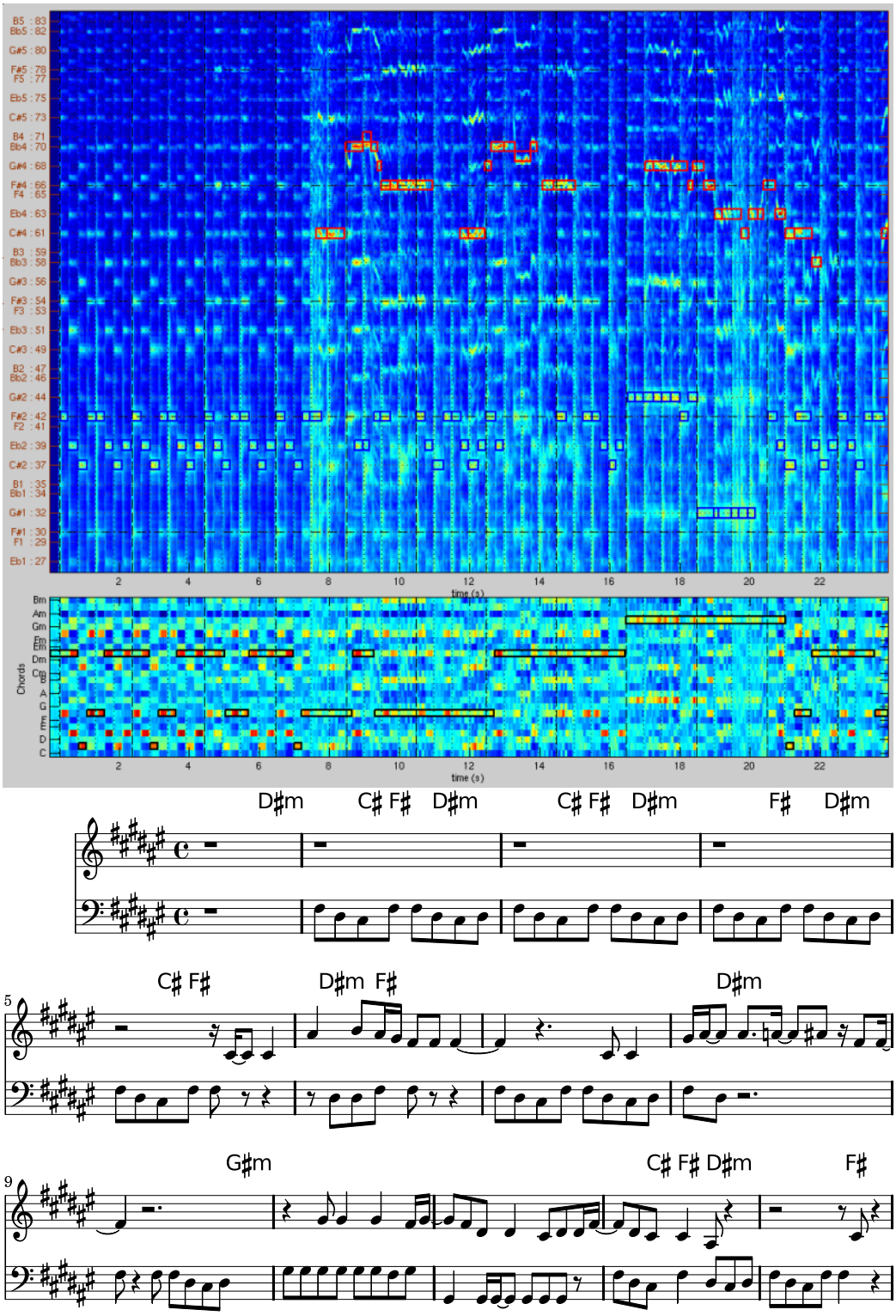


Figure 6.5: An example of the melody, bass line, and chord transcription. The common musical notation is produced directly from the piano-roll representation. See text for details.

gles show the melody notes, the blue rectangles the bass notes, and the black rectangles the chords. Salience function values and the estimated likelihoods for chords are illustrated in the background. In addition, the meter analysis method [64] is used to estimate the temporal positions of measures and beats, shown with dashed and dotted vertical lines, respectively. This temporal grid is used for producing the common musical notation shown in the lower panel. Here the note beginnings and durations are simply quantized to a 16th-note grid although a more elaborate temporal quantization method, such as [134], could be used. The common musical notation is automatically produced by writing the estimated key signature and all the notes and chords with quantized timing into a Lilypond source file. Lilypond[2] is an open-source music typesetting software which compiles text files with a musical notation syntax into musical score documents.

The automatically produced transcription is not perfect as exemplified by Figure 6.5. The song begins with a correctly transcribed monophonic bass line (with drums). However, this is problematic for the chord transcription which follows the bass line notes as the chord roots. In addition, the bass notes D♯, F♯, and C♯ can be interpreted as the D♯m7 chord where the fifth is omitted, or as F♯6 without the third. Therefore, the decision between the triads F♯ and D♯m is difficult even for a human listener. The melody transcription follows the singing melody rather faithfully notating also some glissandi (e.g., at 12.5 s). On the other hand, the predominant singing melody disturbs the bass line transcription, e.g., around 14 s.

Despite the errors, the automatic transcription serves as an excellent starting point for a user who wishes to obtain a perfect transcription of the music piece. With this in mind, the transcription method [P4] has been integrated into music annotation software. Figure 6.6 shows a screenshot of this software called SAMIRAT (Semi-Automatic Music Information Retrieval and Annotation Tool), which has been developed at Tampere University of Technology and now aimed for commercial distribution by a start-up company Wavesum. The software first automatically transcribes a music piece and the user can then edit individual notes or note groups to obtain a perfect transcription. The software allows the simultaneous playback of the music piece and the synthesized transcription, and visualization of the salience function and the results of meter analysis, for example. The software also includes a time-stretching algorithm allowing the user to modify the

[2]Available for download at http://lilypond.org

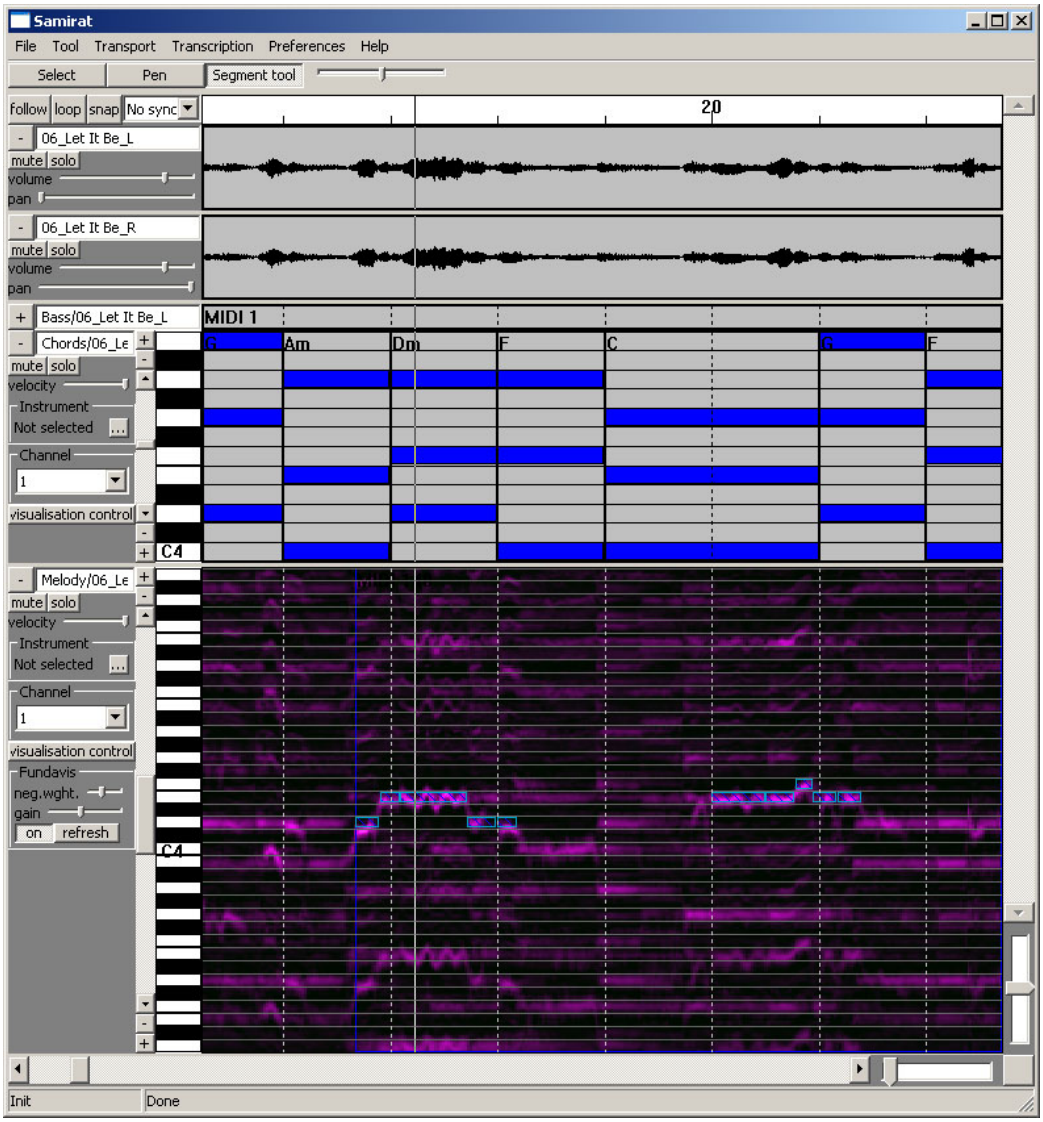


Figure 6.6: A screenshot of SAMIRAT. See text for details.

playback speed in real-time without affecting the pitch. This is a useful feature for a detailed analysis of rapid note passages.

Chapter 7

Applications Based on Automatic Melody Transcription

In addition to automatic or semi-automatic music transcription software, the proposed transcription methods facilitate other applications. This chapter briefly introduces two applications based on the automatic melody transcription method published in [P4]: query by humming of MIDI and audio [P5]; and accompaniment and vocals separation with a karaoke application [P6].

7.1 Query by Humming of MIDI and Audio

Query by humming (QBH) refers to music information retrieval systems where short audio clips of singing or humming act as queries. In a normal use case of QBH, a user wants to find a song from a large database of music recordings. If the user does not remember the name of the artist or the song to make a metadata query, a natural option is to sing, hum, or whistle a part of the melody of the song into a microphone and let the QBH system retrieve the song.

Query by humming has been extensively studied for over a decade [33, 78] and it has remained as an active research topic [14, 122]. QBH systems provide an interesting topic for research as a combination of audio analysis methods, i.e., the automatic singing/humming transcription [27, 48, 142, 115, 132], the study of melodic similarity in symbolic domain [68, 121], and efficient information retrieval techniques [136, 133]. For example, most of the early monophonic singing

transcription methods were developed for QBH systems, as described in [108].

A common approach to QBH first transcribes the user singing either into a F0 trajectory or segmented note events, and then searches for matching melodies from a melody database. Approaches to similarity measurement include string matching techniques [68], hidden Markov models [79, 54], and dynamic programming [53, 133]. The retrieval efficiency is also essential, since for large melody databases, it is not acceptable that the search time depends linearly on the number of database items. For example, Wu *et al.* used a cascade of operations with an increasing similarity requirement to narrow down the search space so that the evaluation of each melody candidate was possible at the final stage [136]. Variants of their system were top-ranked in the MIREX 2006 and 2007 query by singing/humming tasks.

Most of the previous research has concentrated on retrieval from MIDI-melody databases, which are usually manually prepared. However, it is highly desirable to perform the search over audio recordings as well. This allows, for example, the user to perform queries for his or her own music collection. Results on QBH of audio have been reported in [86, 117, 26, 41].

The publication [P5] proposes an efficient QBH system which is capable of performing the search directly from music recordings. The method is briefly introduced in the following but the details are given in [P5]. Given a database of melodies in MIDI format, the method constructs an index of melodic fragments by extracting pitch vectors, where a pitch vector stores an approximate representation of the melody contour within a fixed-length time window. To retrieve audio signals, we use the automatic melody transcription method [P4] to produce the melody database directly from music recordings. The retrieval process converts a sung query into a MIDI note sequence and then extracts pitch vectors from it. For each query pitch vector, the method searches for the nearest neighbors in Euclidean space from the index of database melody fragments to obtain melody candidates and their matching positions in time. This is performed very efficiently by locality sensitive hashing (LSH). The final ranking of candidates is done by comparing the whole transcribed query with each candidate melody segment by using the method by Wu *et al.* [136]. The use of LSH provides a significant speed-up and retrieval performance comparable to the state-of-the-art.

The QBH system [P5] was evaluated in MIREX 2008 "Query by Singing / Humming" task (see [2] for detailed results and abstracts)

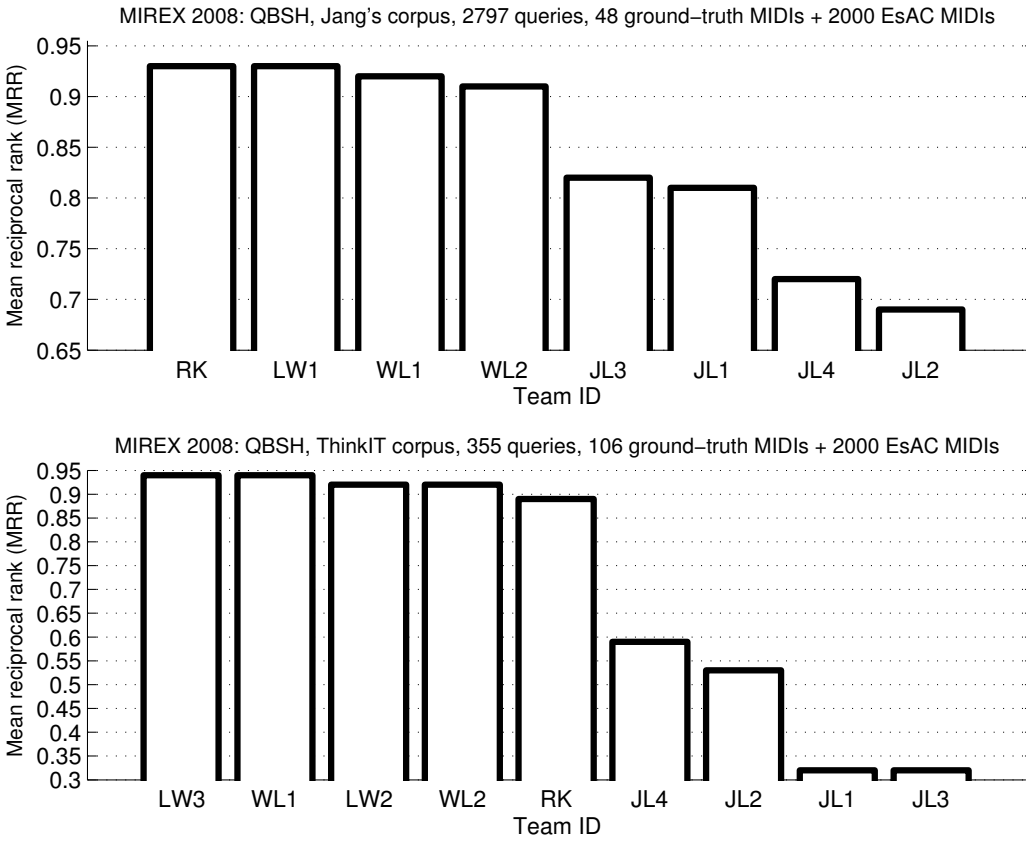


Figure 7.1: MIREX 2008 results of "Query by Singing / Humming" evaluation. The submitted method [P5] is indicated by team "RK".

for which the results are shown in Figure 7.1 for two different corpora. The top panel in Figure 7.1 shows the results for Jang's corpus. The method [P5] indicated by team "RK" was top-ranked together with the submission by L. Wang (team "LW"), where the submissions obtained mean reciprocal rank (MRR) of 0.93. In publication [P5], we evaluated the method with this corpus and reported similar results. The bottom panel in Figure 7.1 shows the results for the ThinkIT corpus. On this corpus, the method obtained MRR of 0.89 whereas the methods by L. Wang and X. Wu & M. Li (team "WL") obtained clearly better results (MRR of 0.94).

The direct search from audio recordings also showed very encouraging results as reported in [P5], where the database of 427 songs contained approximately 26.8 hours of audio. The method retrieved the correct recording in the top-three list for the 58% of 159 hummed

queries. In addition, the proposed method does not require that the user starts the singing from the beginning of the melody, as in [41], for example. This also enables immediate playback of the matched melody segments in the original audio recordings. The automatic transcriptions of some audio recordings still contain too many errors to perform the retrieval robustly. Also the system does not utilize rests, although in some cases, these might be the only cues for retrieval that the user can robustly perform.

7.2 Accompaniment Separation and Karaoke Application

The second application based on the automatic melody transcription method [P4] performs separation of accompaniment and vocals from commercial audio recordings [P6]. In general, music is distributed in a form where all the instruments are mixed together in a monaural or stereophonic audio signal. This type of material is unsuitable for karaoke usage since the lead vocals have been mixed with the accompaniment. Therefore, it is useful to have a method for producing the song accompaniment directly from a desired audio recording.

Sound-source separation has remained as an active research topic for decades. For music signals, upmixing an existing music recording into a representative set of sources (e.g., instruments) or musically meaningful objects, such as notes, enables applications ranging from audio-content modification or remixing (such as the Melodyne software) to structured audio coding, for example. Obtaining such a structured representation is also related to automatic music transcription.

Various techniques have been applied to separate sources in music signals. Ozerov *et al.* [90] adapted a pre-trained generic model of the accompaniment spectra with the detected non-vocal segments in the input sound to separate the vocals and the background. Li and Wang [72] used consecutive steps of vocal/non-vocal segment detection, predominant pitch detection, and the construction of a time-frequency mask for synthesizing the singing melody. Han and Raphael [44] used explicit note information from an automatically-aligned musical score to estimate a binary time-frequency mask to suppress a solo instrument in classical orchestra recordings. Burred and Sikora [7] applied sinusoidal modeling followed by onset detection to form groups of partial

tracks. Each group was then searched for a match in a library of timbre models both to classify and to resynthesize sound sources. Smaragdis and Brown [116] used non-negative spectrogram factorization for separating and transcribing piano notes. Virtanen [129] extended this model with temporal continuity and sparseness criteria to improve the separation performance on music signals. The above-mentioned methods perform the separation for monaural signals, whereas the methods by Barry *et al.* [3], Vinyes *et al.* [128], Vincent [125], and Master [76] utilize the spatial information in stereophonic music signals in the separation.

Here the motivation for using melody transcription is twofold. First, melody transcription produces useful information for a more robust accompaniment separation. The separation can be performed only during the transcribed notes, thus preserving original audio quality during rests. The transcribed notes also allow the robust estimation of a detailed F0 trajectory for the melody which is utilized in the separation. Similar approach to accompaniment and vocals separation has been utilized by Li and Wang [72].

Secondly, karaoke performances are usually given by non-professional singers who may sing out-of-tune, i.e., the singing pitch differs noticeably from the original melody notes. The automatic melody transcription, however, allows the visualization of the original singing melody to aid the singer as in the karaoke game SingStar[1] and in an online karaoke service[2], for example. The transcription can be also used to tune the user singing to the transcribed singing melody in real-time if desired. Nakano *et al.* proposed a method for visualizing the user singing pitch together with the melody analyzed from music recordings [85].

Figure 7.2 shows the GUI of the karaoke-application prototype. The red and green lines show the F0 trajectory of the transcribed melody in the recording and the user singing, respectively. As the figure illustrates, the user sings several semitones too low for the most of the time. The application, however, tunes the user singing to the transcribed melody in real-time if desired. The details of the proposed method and the karaoke application are given in [P6].

The vocals separation of the proposed method has also been utilized in singer identification [81] and in automatic alignment of musical audio and lyrics [80]. Whereas the method [P6] concentrates on the ac-

[1] www.singstargame.com
[2] www.karaokeparty.com

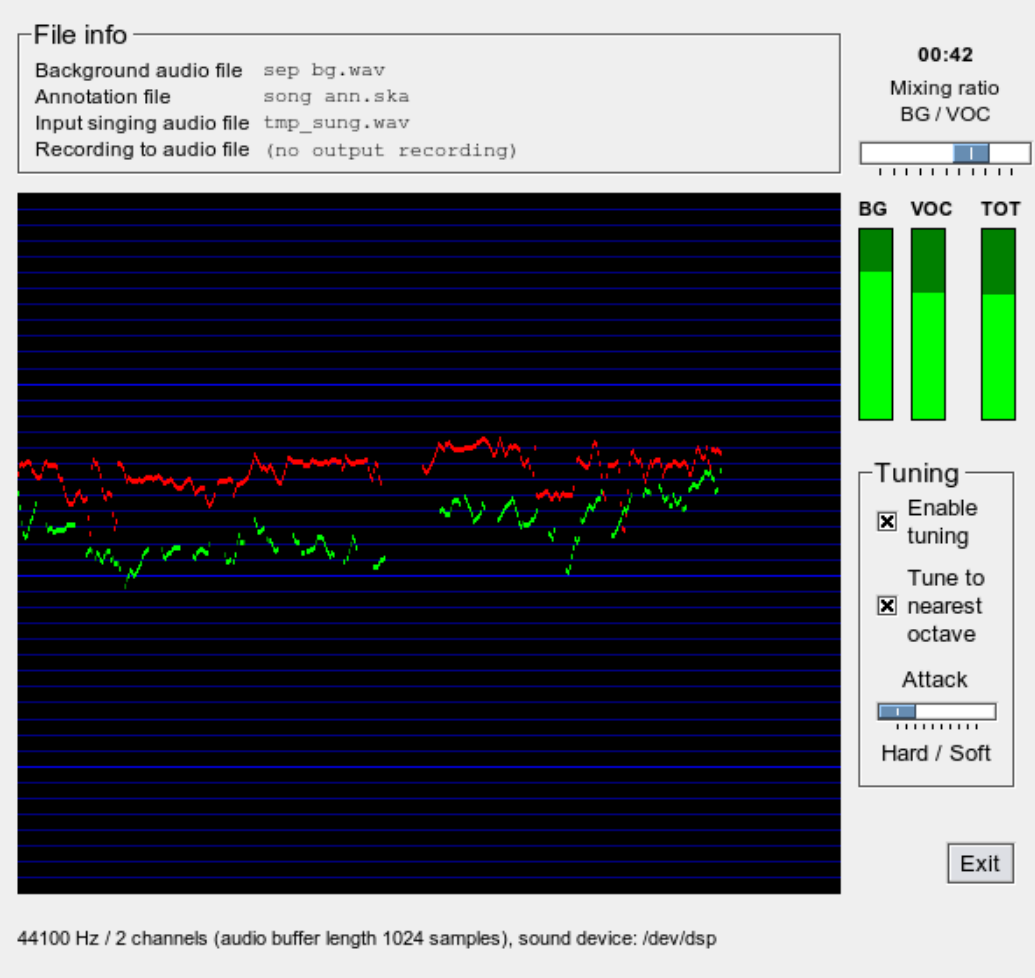


Figure 7.2: Karaoke application with real-time tuning of user singing. The red and green lines show the F0 trajectory of the transcribed melody and the user singing, respectively. See text for details.

companiment separation, the method was further developed to produce better quality for the separated vocals using non-negative spectrogram factorization [131].

Chapter 8

Conclusions and Future Work

The automatic music transcription methods developed in this thesis take an important step towards the transcription of realistic music signals although the problem of automatic music transcription still awaits for a complete solution. The general research focus has shifted from monophonic material to complex polyphonic signals. This trend can also be observed in the work of music-signal processing community, which sets expectations for the upcoming commercial applications aimed at the large audience. This thesis demonstrates that the automatic transcription of complex polyphonic music is possible in practise, and useful applications can be build upon the transcription methods.

The combination of the acoustic and musicological models appears to be a very powerful approach to automatic music transcription. Starting out with monophonic singing transcription, the statistical framework has been successfully applied in polyphonic music transcription with relatively small changes. The framework is very flexible since it allows the models to be trained on any annotated music material, and the individual components, such as the acoustic feature extractors and the key estimation method, can be easily replaced. This is clearly shown by the diversity of the proposed transcription methods. The framework provides an intuitive and practical approach to the music transcription problem.

The three-state HMM represents notes in a very intuitive manner. The note modeling allows to resolve the note segmentation and labeling jointly, whereas most of the other approaches perform these two tasks separately. In literature, the note segmentation has received considerably less attention than the estimation of multiple pitches. This is indicated, e.g., by the results of MIREX 2007 and 2008 evaluations where the differences in the results are more pronounced when also

the note segmentation is required. However, the note segmentation is important and necessary for representing the transcription outputs as MIDI files, for example.

Although the transcription quality still leaves room for improvement, the results are quite satisfactory. From a subjective point of view, the transcription methods already do a better job than musically untrained users. Furthermore, the methods output the absolute note pitches and their temporal positions in music recordings very quickly and provide at least a useful starting point for manual music transcription. Also the two applications based on automatic melody transcription show very promising results.

Automatic music transcription provides interesting research challenges, and there are several possible directions for the future work which have not been investigated in this thesis.

- Utilizing timbre in acoustic modeling. Although the acoustic note models are specifically trained, e.g., for the melody and bass notes, timbre has not been utilized in this work. Features describing the note timbre should be integrated into observation vectors.

- Utilizing stereophonic/multichannel audio. Usually the transcription methods in the literature process only monaural music signals, and to the author's knowledge, there are no transcription methods which utilize stereophonic audio. However, the majority of commercial music is distributed in stereo and multichannel audio has been successfully utilized in sound source separation, for example. This facet should be taken into account also in the transcription methods.

- Top-down approach to music transcription. Most of the transcription methods use the bottom-up approach in music transcription. However, following the human approach to music transcription, top-down processing should be utilized at a large scale. Music recordings repeat the same melody phrases, chord sequences, and sections. A top-down approach could utilize this feature by using music structure analysis to produce similar segments which possibly contain several repetitions of note sequences, for example.

- Utilizing data mining and web resources. There exist large collections of manually prepared MIDI files, printed music, chord maps, and lyrics in the web. Although the collections contain errors and are not and will not be complete, a fair amount of transcriptions

exist for the most popular songs. The integration of automatic music transcription, score alignment, and music-information retrieval techniques can take the next big step in music transcription technology.

Bibliography

[1] Music Information Retrieval Evaluation eXchange (MIREX) 2007. http://www.music-ir.org/mirex/2007, 2007.

[2] Music Information Retrieval Evaluation eXchange (MIREX) 2008. http://www.music-ir.org/mirex/2008, 2008.

[3] D. Barry, B. Lawlor, and E. Coyle. Sound source separation: Azimuth discrimination and resynthesis. In *Proc. 7th International Conference on Digital Audio Effects*, pages 240–244, Naples, Italy, Oct. 2004.

[4] R. Begleiter, R. El-Yaniv, and G. Yona. On prediction using variable order Markov models. *J. of Artificial Intelligence Research*, 22:385–421, 2004.

[5] J. P. Bello and J. Pickens. A robust mid-level representation for harmonic content in music signals. In *Proc. 6th International Conference on Music Information Retrieval*, pages 304–311, 2005.

[6] N. Bertin, R. Badeau, and G. Richard. Blind signal decompositions for automatic transcription of polyphonic music: NMF and K-SVD on the benchmark. In *Proc. 2007 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 65–68, Honolulu, Hawaii, USA, Apr. 2007.

[7] J. J. Burred and T. Sikora. Monaural source separation from musical mixtures based on time-frequency timbre models. In *Proc. 7th International Conference on Music Information Retrieval*, Vienna, Austria, Sept. 2007.

[8] A. T. Cemgil and B. Kappen. Monte Carlo methods for tempo tracking and rhythm quantization. *J. of Artificial Intelligence Research*, 18:45–81, 2003.

[9] A. T. Cemgil, H. J. Kappen, and D. Barber. A generative model for music transcription. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(2):679–694, Mar. 2006.

[10] W. Chai. *Automated Analysis of Musical Structure*. PhD thesis, Massachusetts Institute of Technology, 2005.

[11] W.-C. Chang, A. W. Y. Su, C. Yeh, A. Roebel, and X. Rodet. Multiple-F0 tracking based on a high-order HMM model. In *Proc. 11th International Conference on Digital Audio Effects*, pages 379–386, Espoo, Finland, Sept. 2008.

[12] N. Chétry, M. Davies, and M. Sandler. Musical instrument identification using LSF and K-means. In *Proc. Audio Engineering Society 118th Convention*, Barcelona, Spain, 2005.

[13] R. Dannenberg. An on-line algorithm for real-time accompaniment. In *Proc. International Computer Music Conference*, pages 193–198, 1984.

[14] R. Dannenberg, W. Birmingham, B. Pardo, N. Hu, C. Meek, and G. Tzanetakis. A comparative evaluation of search techniques for query by humming using the MUSART testbed. *J. of the American Society for Information Science and Technology*, 58(3), Feb. 2007.

[15] R. B. Dannenberg and C. Raphael. Music score alignment and computer accompaniment. *Communications of the ACM*, 49(8):38–43, 2006.

[16] M. Davy, S. Godsill, and J. Idier. Bayesian analysis of polyphonic Western tonal music. *J. Acoust. Soc. Am.*, 119(4):2498–2517, Apr. 2006.

[17] A. de Cheveigné. Multiple F0 estimation. In D. L. Wang and G. J. Brown, editors, *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Wiley-IEEE Press, 2006.

[18] A. de Cheveigné and H. Kawahara. Comparative evaluation of F0 estimation algorithms. In *Proc. 7th European Conference on Speech Communication and Technology*, Aalborg, Denmark, 2001.

[19] A. de Cheveign and H. Kawahara. YIN, a fundamental frequency estimator for speech and music. *J. Acoust. Soc. Am.*, 111(4):1917–1930, Apr. 2002.

[20] D. Deutsch, editor. *The Psychology of Music*. Academic Press, second edition, 1999.

[21] S. Dixon. On the computer recognition of solo piano music. In *Proc. Australasian Computer Music Conference*, Brisbane, Australia, 2000.

[22] S. Dixon. Live tracking of musical performances using on-line time warping. In *Proc. 8th International Conference on Digital Audio Effects*, Madrid, Spain, Sept. 2005.

[23] S. Dixon. Evaluation of the audio beat tracking system BeatRoot. *J. of New Music Research*, 36(1):39–50, 2007.

[24] J. S. Downie. The music information retrieval evaluation exchange (2005–2007): A window into music information retrieval research. *Acoustical Science and Technology*, 29(4):247–255, 2008.

[25] K. Dressler. Sinusoidal extraction using an efficient implementation of a multi-resolution FFT. In *Proc. 9th International Conference on Digital Audio Effects*, pages 247–252, Montreal, Canada, Sept. 2006.

[26] A. Duda, A. Nürnberger, and S. Stober. Towards query by humming/singing on audio databases. In *Proc. 7th International Conference on Music Information Retrieval*, Vienna, Austria, Sept. 2007.

[27] A. S. Durey and M. A. Clements. Melody spotting using hidden Markov models. In *Proc. 2nd Annual International Symposium on Music Information Retrieval*, pages 109–117, Oct. 2001.

[28] J. Eggink and G. J. Brown. Extracting melody lines from complex audio. In *Proc. 5th International Conference on Music Information Retrieval*, 2004.

[29] D. Ellis and G. Poliner. Classification-based melody transcription. *Machine Learning Journal*, 65(2–3):439–456, 2006.

[30] D. FitzGerald and J. Paulus. Unpitched percussion transcription. In Klapuri and Davy [63], pages 131–162.

[31] G. D. Forney. The Viterbi algorithm. *Proc. IEEE*, 61(3):268–278, Mar. 1973.

[32] H. Fujihara, T. Kitahara, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno. F0 estimation method for singing voice in polyphonic audio signal based on statistical vocal model and Viterbi search. In *Proc. 2006 IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 2006.

[33] A. Ghias, J. Logan, and D. Chamberlin. Query by humming: Musical information retrieval in an audio database. In *Proc. ACM Multimedia Conference '95*, San Fransisco, California, Nov. 1995. Cornell University.

[34] E. Gómez. *Tonal Description of Music Audio Signals*. PhD thesis, Universitat Pompeu Fabra, Barcelona, Spain, 2006.

[35] M. Goto. A robust predominant-F0 estimation method for real-time detection of melody and bass lines in CD recordings. In *Proc. 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 757–760, June 2000.

[36] M. Goto. A real-time music-scene-description system: predominant-F0 estimation for detecting melody and bass lines in real-world audio signals. *Speech Communication*, 43(4):311–329, 2004.

[37] M. Goto. Music scene description. In Klapuri and Davy [63], pages 327–359.

[38] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Popular, classical, and jazz music databases. In *Proc. 3rd International Conference on Music Information Retrieval*, pages 287–288, 2002.

[39] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Music genre database and musical instrument sound database. In *Proc. 4th International Conference on Music Information Retrieval*, 2003.

[40] F. Gouyon. *A computational approach to rhythm description – Audio features for the computation of rhythm periodicity functions and their use in tempo induction and music content processing*. PhD thesis, Universitat Pompeu Fabra, Barcelona, Spain, 2005.

[41] L. Guo, X. He, Y. Zhang, and Y. Lu. Content-based retrieval of polyphonic music objects using pitch contour. In *Proc. 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 2205–2208, Las Vegas, Nevada, USA, Apr. 2008.

[42] S. Hainsworth. Beat tracking and musical metre analysis. In Klapuri and Davy [63], pages 101–129.

[43] S. W. Hainsworth and M. D. Macleod. Automatic bass line transcription from polyphonic music. In *Proc. International Computer Music Conference*, pages 431–434, 2001.

[44] Y. Han and C. Raphael. Desoloing monaural audio using mixture models. In *Proc. 7th International Conference on Music Information Retrieval*, pages 145–148, Vienna, Austria, Sept. 2007.

[45] C. Harte, M. Sandler, S. Abdallah, and E. Gómez. Symbolic representation of musical chords: a proposed syntax for text annotations. In *Proc. 6th International Conference on Music Information Retrieval*, pages 66–71, 2005.

[46] C. A. Harte and M. B. Sandler. Automatic chord identification using a quantised chromagram. In *Proc. 118th Audio Engineering Society's Convention*, 2005.

[47] W. M. Hartmann. Pitch, periodicity, and auditory organization. *J. Acoust. Soc. Am.*, 100(6):3491–3502, Dec. 1996.

[48] G. Haus and E. Pollastri. An audio front end for query-by-humming systems. In *Proc. 2nd Annual International Symposium on Music Information Retrieval*, Oct. 2001.

[49] P. Herrera-Boyer, A. Klapuri, and M. Davy. Automatic classification of pitched musical instrument sounds. In Klapuri and Davy [63], pages 163–200.

[50] W. J. Hess. Pitch and voicing determination. In S. Furui and M. M. Sondhi, editors, *Advances in speech signal processing*, pages 3–48. Marcel Dekker, Inc., New York, 1991.

[51] X. Huang, A. Acero, and H.-W. Hon. *Spoken Language Processing: A Guide to Theory, Algorithm and System Development*. Prentice Hall, 2001.

[52] Ö. İzmirli. Audio key finding using low-dimensional spaces. In *Proc. 7th International Conference on Music Information Retrieval*, pages 127–132, 2006.

[53] J.-S. R. Jang and M.-Y. Gao. A query-by-singing system based on dynamic programming. In *Proc. International Workshop on Intelligent Systems Resolutions*, 2000.

[54] J.-S. R. Jang, C.-L. Hsu, and H.-R. Lee. Continuous HMM and its enhancement for singing/humming query retrieval. In *Proc. 6th International Conference on Music Information Retrieval*, 2005.

[55] J.-S. R. Jang, H.-R. Lee, and C.-H. Yeh. Query by tapping: A new paradigm for content-based music retrieval from acoustic input. In *Proc. Proceedings of the Second IEEE Pacific Rim Conference on Multimedia: Advances in Multimedia Information Processing*, pages 590–597, 2001.

[56] M. Karjalainen. Sound quality measurements of audio systems based on models of auditory perception. In *Proc. 1984 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 132–135, Mar. 1984.

[57] K. Kashino. Auditory scene analysis in music signals. In Klapuri and Davy [63], pages 299–325.

[58] K. Kashino, K. Nakadai, T. Kinoshita, and H. Tanaka. Application of Bayesian probability network to music scene analysis. In *Working Notes of IJCAI Workshop of Computational Auditory Scene Analysis (IJCAI-CASA)*, pages 52–59, Aug. 1995.

[59] S. Z. K. Khine, T. L. Nwe, and H. Li. Singing voice detection in pop songs using co-training algorithm. In *Proc. 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 1629–1632, Las Vegas, Nevada, USA, Apr. 2008.

[60] A. Klapuri. A perceptually motivated multiple-F0 estimation method. In *Proc. 2005 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 291–294, New Paltz, NY, USA, Oct. 2005.

[61] A. Klapuri. Multiple fundamental frequency estimation by summing harmonic amplitudes. In *Proc. 7th International Conference on Music Information Retrieval*, pages 216–221, 2006.

[62] A. Klapuri. Multipitch analysis of polyphonic music and speech signals using an auditory model. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2):255–266, Feb. 2008.

[63] A. Klapuri and M. Davy, editors. *Signal Processing Methods for Music Transcription*. Springer Science + Business Media LLC, 2006.

[64] A. P. Klapuri, A. J. Eronen, and J. T. Astola. Analysis of the meter of acoustic musical signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1):342–355, Jan. 2006.

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

[66] O. Lartillot, S. Dubnov, G. Assayag, and G. Bejerano. Automatic modeling of musical style. In *Proc. International Computer Music Conference*, 2001.

[67] K. Lee. *A System for Acoustic Chord Transcription and Key Extraction from Audio Using Hidden Markov Models Trained on Synthesized Audio*. PhD thesis, Stanford University, Mar. 2008.

[68] K. Lemström. *String Matching Techniques for Music Retrieval*. PhD thesis, University of Helsinki, 2000.

[69] F. Lerdahl and R. Jackendoff. *A Generative Theory of Tonal Music*. MIT Press, 1983.

[70] M. Levy and M. Sandler. Structural segmentation of musical audio by constrained clustering. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2):318–326, Feb. 2008.

[71] Y. Li and D. L. Wang. Pitch detection in polyphonic music using instrument tone models. In *Proc. 2007 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 481–484, Honolulu, Hawaii, USA, Apr. 2007.

[72] Y. Li and D. L. Wang. Separation of singing voice from music accompaniment for monaural recordings. *IEEE Transactions on*

Audio, Speech, and Language Processing, 15(4):1475–1487, May 2007.

[73] R. Lyon. A computational model of filtering, detection, and compression in the cochlea. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 1282–1285, May 1982.

[74] M. Marolt. A connectionist approach to automatic transcription of polyphonic piano music. *IEEE Transactions on Multimedia*, 6(3):439–449, 2004.

[75] M. Marolt. Gaussian mixture models for extraction of melodic lines from audio recordings. In *Proc. 5th International Conference on Music Information Retrieval*, 2004.

[76] A. S. Master. *Stereo Music Source Separation Via Bayesian Modeling*. PhD thesis, Stanford University, 2006.

[77] M. F. McKinney, D. Moelants, M. E. P. Davies, and A. Klapuri. Evaluation of audio beat tracking and music tempo extraction algorithms. *J. of New Music Research*, 36(1):1–16, 2007.

[78] R. McNab, L. Smith, I. Witten, C. Henderson, and S. Cunningham. Towards the digital music library: Tune retrieval from acoustic input. In *Proc. First ACM International Conference on Digital Libraries*, pages 11–18, 1996.

[79] C. Meek and W. Birmingham. Applications of binary classification and adaptive boosting to the query-by-humming problem. In *Proc. 3rd International Conference on Music Information Retrieval*, 2002.

[80] A. Mesaros and T. Virtanen. Automatic alignment of music audio and lyrics. In *Proc. 11th International Conference on Digital Audio Effects*, pages 321–324, Espoo, Finland, Sept. 2008.

[81] A. Mesaros, T. Virtanen, and A. Klapuri. Singer identification in polyphonic music using vocal separation and pattern recognition methods. In *Proc. 7th International Conference on Music Information Retrieval*, pages 375–378, Vienna, Austria, Sept. 2007.

[82] K. Miyamoto, H. Kameoka, H. Takeda, T. Nishimoto, and S. Sagayama. Probabilistic approach to automatic music transcription from audio signals. In *Proc. 2007 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 697–700, Honolulu, Hawaii, USA, Apr. 2007.

[83] J. A. Moorer. On the transcription of musical sound by computer. *Computer Music Journal*, 1(4):32–38, Nov. 1977.

[84] K. P. Murphy. *Dynamic Bayesian Networks: Representation, Inference, and Learning*. PhD thesis, University of California, Berkeley, 2002.

[85] T. Nakano, M. Goto, and Y. Hiraga. MiruSinger: A singing skill visualization interface using real-time feedback and music CD recordings as referential data. In *Proc. Ninth IEEE International Symposium on Multimedia Workshops*, pages 75–76, Dec. 2007.

[86] T. Nishimura, H. Hashiguchi, J. Takita, J. X. Zhang, M. Goto, and R. Oka. Music signal spotting retrieval by a humming query using start frame feature dependent continuous dynamic programming. In *Proc. 2nd Annual International Symposium on Music Information Retrieval*, pages 211–218, Oct. 2001.

[87] K. Noland and M. Sandler. Key estimation using a hidden Markov model. In *Proc. 7th International Conference on Music Information Retrieval*, pages 121–126, 2006.

[88] B. S. Ong. *Structural Analysis and Segmentation of Music Signals*. PhD thesis, Universitat Pompeu Fabra, Barcelona, Spain, 2006.

[89] N. Orio and M. S. Sette. A HMM-based pitch tracker for audio queries. In *Proc. 4th International Conference on Music Information Retrieval*, pages 249–250, 2003.

[90] A. Ozerov, P. Philippe, F. Bimbot, and R. Gribonval. Adaptation of Bayesian models for single-channel source separation and its application to voice/music separation in popular songs. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(5):1564–1578, July 2007.

[91] F. Pachet. The Continuator: Musical interaction with style. *J. of New Music Research*, 32(3):333–341, Sept. 2003.

[92] R. P. Paiva. *Melody Detection in Polyphonic Audio*. PhD thesis, University of Coimbra, 2006.

[93] H. Papadopoulos and G. Peeters. Large-scale study of chord estimation algorithms based on chroma representation and HMM. In *Proc. International Workshop on Content-Based Multimedia Indexing*, pages 53–60, June 2007.

[94] J. Paulus and A. Klapuri. Music structure analysis using a probabilistic fitness measure and an integrated musicological model. In *Proc. 9th International Conference on Music Information Retrieval*, pages 369–374, Philadelphia, Pennsylvania, USA, Sept. 2008.

[95] G. Peeters. Chroma-based estimation of musical key from audio-signal analysis. In *Proc. 7th International Conference on Music Information Retrieval*, pages 115–120, 2006.

[96] G. Peeters and X. Rodet. Hierarchical Gaussian tree with inertia ratio maximization for the classification of large musical instrument databases. In *Proc. International Conference on Digital Audio Effects*, 2003.

[97] A. Pertusa and J. M. Iñesta. Multiple fundamental frequency estimation using Gaussian smoothness. In *Proc. 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 105–108, Las Vegas, Nevada, USA, Apr. 2008.

[98] G. Poliner, D. Ellis, A. Ehmann, E. Gómez, S. Streich, and B. Ong. Melody transcription from music audio: Approaches and evaluation. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(4):1247–1256, May 2007.

[99] G. E. Poliner and D. P. W. Ellis. A discriminative model for polyphonic piano transcription. *EURASIP Journal on Applied Signal Processing*, 2007.

[100] H. Purwins. *Profiles of Pitch Classes – Circularity of Relative Pitch and Key: Experiments, Models, Music Analysis, and Perspectives*. PhD thesis, Berlin University of Technology, 2005.

[101] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, 77(2):257–289, Feb. 1989.

[102] C. Raphael. Automatic segmentation of acoustic musical signals using hidden Markov models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4):360–370, Apr. 1999.

[103] C. Raphael. A probabilistic expert system for automatic musical accompaniment. *J. of Computational and Graphical Statistics*, 10(3):487–512, Sept. 2001.

[104] C. Raphael. Automatic transcription of piano music. In *Proc. 3rd International Conference on Music Information Retrieval*, pages 15–19, 2002.

[105] L. Rossi and G. Girolami. Identification of polyphonic piano signals. *ACUSTICA acta acustica*, 83:1077–1084, 1997.

[106] T. D. Rossing. *The Science of Sound*. Addison-Wesley Publishing Company Inc., second edition, 1990.

[107] M. Ryynänen. Probabilistic modelling of note events in the transcription of monophonic melodies. Master's thesis, Tampere University of Technology, Tampere, Finland, Mar. 2004.

[108] M. Ryynänen. Singing transcription. In Klapuri and Davy [63], pages 361–390.

[109] M. Ryynänen and A. Klapuri. Modelling of note events for singing transcription. In *Proc. ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio*, Oct. 2004.

[110] M. Ryynänen, T. Virtanen, J. Paulus, and A. Klapuri. Method for processing melody. Finnish patent application no. 20075737, Oct. 2007.

[111] E. D. Scheirer. *Music-Listening Systems*. PhD thesis, School of Architecture and Planning, MIT Media Laboratory, 2000.

[112] J. Seppänen, A. Eronen, and J. Hiipakka. Joint beat & tatum tracking from music signals. In *Proc. 7th International Conference on Music Information Retrieval*, pages 23–28, 2006.

[113] A. Sheh and D. P. Ellis. Chord segmentation and recognition using EM-trained hidden Markov models. In *Proc. 4th International Conference on Music Information Retrieval*, 2003.

[114] H.-H. Shih, S. S. Narayanan, and C.-C. J. Kuo. An HMM-based approach to humming transcription. In *Proc. 2002 IEEE International Conference on Multimedia and Expo*, Aug. 2002.

[115] H.-H. Shih, S. S. Narayanan, and C.-C. J. Kuo. Multidimensional humming transcription using a statistical approach for query by humming systems. In *Proc. 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 541–544, Apr. 2003.

[116] P. Smaragdis and J. C. Brown. Non-negative matrix factorization for polyphonic music transcription. In *Proc. 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 177–180, New Paltz, NY, USA, Oct. 2003.

[117] J. Song, B. S. Y., and K. Yoon. Query by humming: matching humming query to polyphonic audio. In *Proc. 2002 IEEE International Conference on Multimedia and Expo*, pages 329–332, Aug. 2002.

[118] D. Temperley. *The Cognition of Basic Musical Structures*. MIT Press, 2001.

[119] D. Temperley and D. Sleator. Modeling meter and harmony: A preference-rule approach. *Computer Music Journal*, 23(1):10–27, 1999.

[120] P. Toiviainen. An interactive MIDI accompanist. *Computer Music Journal*, 22(4):63–75, 1998.

[121] R. Typke. *Music Retrieval based on Melodic Similarity*. PhD thesis, Universiteit Utrecht, 2007.

[122] E. Unal, E. Chew, P. G. Georgiou, and S. S. Narayanan. Challenging uncertainty in query by humming systems: A fingerprinting approach. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2):359–371, Feb. 2008.

[123] B. L. Vercoe, W. G. Gardner, and E. D. Scheirer. Structured audio: creation, transmission, and rendering of parametric sound representations. *Proc. IEEE*, 86(5):922–940, May 1998.

[124] T. Viitaniemi, A. Klapuri, and A. Eronen. A probabilistic model for the transcription of single-voice melodies. In *Proc. 2003 Finnish Signal Processing Symposium*, pages 59–63, May 2003.

[125] E. Vincent. Musical source separation using time-frequency source priors. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1):91–98, Jan. 2006.

[126] E. Vincent, N. Bertin, and R. Badeau. Harmonic and inharmonic nonnegative matrix factorization for polyphonic pitch transcription. In *Proc. 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 109–112, Las Vegas, Nevada, USA, Apr. 2008.

[127] E. Vincent and M. D. Plumbley. Low-bitrate object coding of musical audio using Bayesian harmonic models. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(4):1273–1282, May 2007.

[128] M. Vinyes, J. Bonada, and A. Loscos. Demixing commercial music productions via human-assisted time-frequency masking. In *Proc. Audio Engineering Society 120th Convention*, Paris, France, May 2006.

[129] T. Virtanen. Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(3):1066–1074, Mar. 2006.

[130] T. Virtanen. *Sound Source Separation in Monaural Music Signals*. PhD thesis, Tampere University of Technology, Tampere, Finland, 2006.

[131] T. Virtanen, A. Mesaros, and M. Ryynänen. Combining pitch-based inference and non-negative spectrogram factorization in separating vocals from polyphonic music. In *Proc. ISCA Tutorial and Research Workshop on Statistical and Perceptual Audition*, Brisbane, Australia, Sept. 2008.

[132] C. Wang, R. Lyu, and Y. Chiang. A robust singing melody tracker using adaptive round semitones (ARS). In *Proc. 3rd International Symposium on Image and Signal Processing and Analysis*, pages 549–554, 2003.

[133] L. Wang, S. Huang, S. Hu, J. Liang, and B. Xu. An effective and efficient method for query by humming system based on multi-similarity measurement fusion. In *Proc. International Confer-*

ence on Audio, Language and Image Processing, pages 471–475, July 2008.

[134] N. Whiteley, A. T. Cemgil, and S. Godsill. Bayesian modelling of temporal structure in musical audio. In *Proc. 7th International Conference on Music Information Retrieval*, pages 29–34, 2006.

[135] I. H. Witten and T. C. Bell. The zero-frequency problem: estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory*, 37(4):1085–1094, July 1991.

[136] X. Wu, M. Li, J. Yang, and Y. Yan. A top-down approach to melody match in pitch countour for query by humming. In *Proc. International Conference of Chinese Spoken Language Processing*, 2006.

[137] C. Yeh. *Multiple Fundamental Frequency Estimation of Polyphonic Recordings*. PhD thesis, Université Paris VI, 2008.

[138] K. Yoshii, M. Goto, and H. G. Okuno. Drum sound recognition for polyphonic audio signals by adaptation and matching of spectrogram templates with harmonic structure suppression. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(1):333–345, Jan. 2007.

[139] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. *The HTK Book (for HTK version 3.4)*. Cambridge University Engineering Department, Dec. 2006.

[140] S. J. Young, N. H. Russell, and J. H. S. Thornton. Token passing: a simple conceptual model for connected speech recognition systems. Technical report, Cambridge University Engineering Department, July 1989.

[141] R. Zhou. *Feature Extraction of Musical Content for Automatic Music Transcription*. PhD thesis, University of Lausanne, Switzerland, 2006.

[142] Y. Zhu and M. S. Kankanhalli. Robust and efficient pitch tracking for query-by-humming. In *Proc. 2003 Joint Conference of the Fourth International Conference on Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim*

Conference on Multimedia, volume 3, pages 1586–1590, Dec. 2003.

Errata and Clarifications for the Publications

- [P1], page 321, Section 3.3. "A transition from the silence model to itself is prohibited." is a bit confusing; there is no silence-to-silence transition in the musicological model but the silence model itself (a one-state HMM) has a self-transition.
- [P2], Table 3. The "Distance on the circle of fifths" in the last column should be > 3, not ≥ 3.

Publication P1

M. Ryynänen and A. Klapuri, "Polyphonic music transcription using note event modeling," in *Proc. 2005 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, (New Paltz, New York, USA), pp. 319–322, Oct. 2005.

Publication P2

M. Ryynänen and A. Klapuri, "Transcription of the singing melody in polyphonic music," in *Proc. 7th International Conference on Music Information Retrieval*, (Victoria, Canada), pp. 222–227, Oct. 2006.

Publication P3

M. Ryynänen and A. Klapuri, "Automatic bass line transcription from streaming polyphonic audio," in *Proc. 2007 IEEE International Conference on Acoustics, Speech, and Signal Processing*, (Honolulu, Hawaii, USA), pp. 1437–1440, Apr. 2007.

Publication P4

M. Ryynänen and A. Klapuri, "Automatic transcription of melody, bass line, and chords in polyphonic music," *Computer Music Journal*, 32:3, pp. 72–86, Fall 2008.

Publication P5

M. Ryynänen and A. Klapuri, "Query by humming of MIDI and audio using locality sensitive hashing," in *Proc. 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing*, (Las Vegas, Nevada, USA), pp. 2249–2252, Apr. 2008.

Publication P6

M. Ryynänen, T. Virtanen, J. Paulus, and A. Klapuri, "Accompaniment separation and karaoke application based on automatic melody transcription," in *Proc. 2008 IEEE International Conference on Multimedia and Expo*, (Hannover, Germany), pp. 1417–1420, June 2008.