A User-assisted Approach to Multiple Instrument Music Transcription

Holger Kirchhoff

PhD thesis

Submitted in partial fulfillment of the requirements
of the Degree of Doctor of Philosophy

School of Electronic Engineering and Computer Science
Queen Mary University of London

2013

I, Holger Kirchhoff, confirm that the research included within this thesis is my own work or that where it has been carried out in collaboration with, or supported by others, that this is duly acknowledged below and my contribution indicated. Previously published material is also acknowledged below.

I attest that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge break any UK law, infringe any third party's copyright or other Intellectual Property Right, or contain any confidential material.

I accept that the College has the right to use plagiarism detection software to check the electronic version of the thesis.

I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university.

Signature:
Date:

Details of collaboration and publications:
A list of publications and the author's contributions therein can be found in Section 1.4.

Abstract

The task of automatic music transcription has been studied for several decades and is regarded as an enabling technology for a multitude of applications such as music retrieval and discovery, intelligent music processing and large-scale musicological analyses. It refers to the process of identifying the musical content of a performance and representing it in a symbolic format. Despite its long research history, fully automatic music transcription systems are still error prone and often fail when more complex polyphonic music is analysed. This gives rise to the question in what ways human knowledge can be incorporated in the transcription process.

This thesis investigates ways to involve a human user in the transcription process. More specifically, it is investigated how user input can be employed to derive timbre models for the instruments in a music recording, which are employed to obtain instrument-specific (parts-based) transcriptions.

A first investigation studies different types of user input in order to derive instrument models by means of a non-negative matrix factorisation framework. The transcription accuracy of the different models is evaluated and a method is proposed that refines the models by allowing each pitch of each instrument to be represented by multiple basis functions.

A second study aims at limiting the amount of user input to make the method more applicable in practice. Different methods are considered to estimate missing non-negative basis functions when only a subset of basis functions can be extracted based on the user information.

A method is proposed to track the pitches of individual instruments over time by means of a Viterbi framework in which the states at each time frame contain several candidate instrument-pitch combinations. A transition probability is employed that combines three different criteria: the frame-wise reconstruction error of each combination, a pitch continuity measure that favours similar pitches in consecutive frames, and an explicit activity model for each instrument. The method is shown to outperform other state-of-the-art multi-instrument tracking methods.

Finally, the extraction of instrument models that include phase information is investigated as a step towards complex matrix decomposition. The phase relations between the partials of harmonic sounds are explored as a time-invariant property that can be employed to form complex-valued basis functions. The application of the model for a user-assisted transcription task is illustrated with a saxophone example.

Acknowledgements

Time flies. Three years as a research student have passed and it feels as if I have only just started. During this time, I had the opportunity to meet and work with so many knowledgeable and inspiring people and I am grateful for the support of many in many ways.

Primarily, I would like to express my gratitude to my supervisors Simon Dixon and Anssi Klapuri. I enjoyed three years of substantiated guidance, insightful discussions and personal encouragement. Both have substantially shaped my research and at the same time provided me the opportunity and the freedom to develop and test my own ideas. I would also like to thank my independent assessor Mark Plumbley who looked over my work at the different stages of the programme.

I enjoyed collaborating with several people on joint publications. Particularly I would like to thank Roland Badeau, whose excellence in all mathematical aspects was invaluable for our project on phase-based instrument models. A special thanks also to Emmanouil Benetos and Dimitrios Giannoulis for discussions and brainstorming about the future of research on transcription systems.

I am also grateful to many people in the Media and Arts Technology programme, particularly those who started this journey with me. With their incredibly diverse backgrounds, these people have broadened my horizon and my understanding of technology in many different ways. Thank you Saul Albert, Pollie Barden, Ben Bengler, Ilze Black, Audhild Dahlstrøm, Henrik Ekeus, Dominic Freeston, Berit Greinke, Toby Harris, Sara Heitlinger, Katja Knecht, Phillip Neil Martin, Antonella Mazzoni, David Meckin, Duncan Menzies, Nicola Plant, Grzesiek Sedek and Keir Williams. A special thanks goes also to Richard Kelly without whom the MAT programme would not be what it is now. Thanks also to Mark Sandler and Pat Healey for accepting me on the programme and to the College for providing the funding.

A whole lot of other people made my stay at the Centre for Digital Music enjoyable and worthwhile. I tremendously enjoyed the countless lunch breaks, lots of good laughs and sometimes philosophical discussions with Sebastian Ewert,

Marco Fabiani, Luis Figueira, Joachim Fritsch, Steven Hargreaves, Martín Hartmann, Elio Quinton and David Ronan. Further thanks to other members and visitors of C4DM: Daniele Barchiesi, Mathieu Barthet, Chris Cannam, Tian Cheng, Magdalena Chudy, Alice Clifford, Gyorgy Fazekas, Peter Foster, Joachim Ganseman, Ken O'Hanlon, Katerina Kosta, Matthias Mauch, Brecht De Man, Andrew Robertson, Sebastian Schlecht, Siddarth Sigtia, Jordan Smith, Chunyang Song, Yading Song, Dan Stowell, Bob Sturm, Mi Tian, Mike Terrell, Rob Tubb, Bogdan Vera and Sonia Wilkie.

I would not have been able to carry out this research without the love and support of my wife Nina. Thank you for cheering me up countless times, getting my mind off my work, helping me through difficult times and simply for being wonderful. We had an amazing time here in London and in the UK, and we tremendously enjoyed discovering the diversity, beauty, and all the little quirks of this island and its people.

A very big thank you also goes to my parents. Without your support I would not have been able to pile up all that knowledge in order to get to the point of pursuing a PhD in the first place.

Contents

List of Figures

List of Abbreviations

ACF	Autocorrelation function
AMT	Automatic music transcription
AR	Autoregressive
ASA	Auditory scene analysis
BP	Basis pursuit
CASA	Computational auditory scene analysis
DFT	Discrete Fourier transform
EM	Expectation maximisation
ERB	Equivalent rectangular bandwidth
f0	Fundamental frequency
HMM	Hidden Markov model
HTC	Harmonic temporal structured clustering
i.i.d.	Independent and identically distributed
ICA	Independent component analysis
IRLS	Iterative reweighted least squares
IS	Itakura-Saito
ISA	Independent subspace analysis
KL	Kullback-Leibler
LS	Least-squares
MA	Moving average
MAP	Maximum-a-posteriori
MAPS	MIDI aligned piano sounds
MCMC	Markov-Chain-Monte-Carlo
MF0E	Multiple-f0 estimation
MIDI	Musical instrument digital interface
MIREX	Music Information Retrieval Evaluation eXchange
ML	Maximum likelihood
MLP	Multilayer perceptron
MP	Matching pursuit
NMF	Non-negative matrix factorisation

NMF2D	Non-negative matrix factor 2-D deconvolution
NMFD	Non-negative matrix factor deconvolution
NT	Note tracking
OMP	Orthogonal matching pursuit
PCA	Principal component analysis
PLCA	Probabilistic latent component analysis
RBF	Radial basis function
RTFI	Resonator time frequency image
SACF	Summary autocorrelation function
siNMF	Shift-invariant NMF
SVM	Support vector machine
TDNN	Time-delay neural network
TWM	Two-way mismatch

List of Symbols

$\bullet$	elementwise multiplication of matrices or vectors
$.^\top$	matrix or vector transposition
$\lfloor \cdot \rceil$	rounding to the nearest integer
$\hat{\cdot}$	parameter estimate
$\angle \cdot$	phase angle
$\lvert \cdot \rvert$	modulus
$a, \overline{a}$	activity and inactivity flag
a_p	amplitude of partial p
β	parameter for β-divergence
$B(n,k)$	complex spectrogram of a monophonic instrument
c	constant
C_β	β-divergence cost function
CP_i	combined precision of instrument i
d	index of provided spectra
$d_\beta(x,y)$	β-divergence of x and y
$\delta_n(x)$	probability of the most likely Viterbi state sequence that ends in state S_x in frame n
D	number of provided spectra
$\Delta\varphi_p$	phase offset between p-th partial and fundamental
e	reconstruction error
$\mathbf{e}$	excitation spectrum of source-filter model
$f(n)$	phase based activity measure
$\mathbf{f}$	adaptation filter
FN	number of false negatives
FP	number of false positives
g	single gain value
$g(n)$	time-varying gain factor
$\mathbf{G}$	combined pitch activation matrix of all instruments
$\mathbf{G^i}$	pitch activation matrix of instrument i
$h(t)$	window function

$\mathbf{h}$	filter spectrum of source-filter model
$H(\Omega)$	window spectrum
$\mathbf{H}$	matrix of activations/gains
i	instrument index
I	number of instruments
IP_i	instrument precision of instrument i
k	frequency index
K	number of frequency bins
$\mathbf{\Lambda}$	non-negative approximation of magnitude spectrogram
m	hop size
M	number of peaks
n	time frame index
N	number of time frames
N_S	number of Viterbi states per frame
ω_p	angular frequency of partial p
O	observation sequence
Ω_k	normalised angular frequency of the k-th frequency index
p	partial index
$p(\cdot)$	probability
p_a	probability of instrument activity
p_d	probability of pitch continuity
p_e	probability based on reconstruction error
p_k	partial index of the k-th frequency index
φ_p	instantaneous phase of partial p
$\psi_n(x)$	state index at frame $n-1$ of most likely state sequence that ends in state S_x in frame n
P	number of partials
PP_i	pitch precision of instrument i
q_n	Viterbi state at time n
$q(n,k)$	phase ambiguity term
r	basis function/template index
R	number of basis functions/templates
$s(t)$	harmonic signal
$\mathbf{s}$	vector of scaling factors for source-filter model
σ_d	standard deviation of p_d
σ_e	standard deviation of p_e
S_x	Viterbi state with index x
$S(n,k)$	STFT of a harmonic signal
$S'(n,k)$	simplified STFT of a harmonic signal
t	time index

τ	time shift
$\theta(t)$	instantaneous phase of fundamental at time t
$\mathcal{T}$	number of time shifts
TN	number of true negatives
TP	number of true positives
TPC	number of correctly detected pitches among true positives
$\Theta(n)$	instantaneous phase of fundamental at frame n
$\Theta_u(n)$	unwrapped instantaneous phase of fundamental at frame n
$V(n,k)$	complex spectrogram of a mixture
$\mathbf{V}$	magnitude spectrogram
$\mathbf{w}$	basis function/template vector
$\mathbf{W}$	matrix of basis functions/templates
$x(t)$	arbitrary signal
$X(n,k)$	STFT of arbitrary signal

Chapter 1

Introduction

1.1 Motivation

The introduction of digital audio formats in the early 1980s not only had a substantial impact on the recording industry and the way music is stored, distributed and consumed today, it also opened up new possibilities for accessing, analysing and manipulating recorded music. In combination with the emergence of personal digital computing devices, it gave rise to a whole new family of computational algorithms that aim at automatically analysing aspects of the musical content of a digital recording. Music analysis and processing algorithms have since been beneficial for various groups within the music industry: recording engineers and producers benefit from algorithms that analyse and manipulate recorded music in various new ways and with a precision that could not be accomplished with analog devices; consumers seek to retrieve music tracks from large music collections based on automatically extracted music content; musicians use analysis algorithms to guide their instrumental practice and similarly musicologists are interested in detailed analyses of various musical aspects. The common ground for these practical use cases is the extraction of semantic information about the music content from the recording itself.

Automatic music transcription (AMT) is one of the major tasks within this set of algorithms. It aims at finding an algorithmic formulation for the process of music transcription. In a musicological sense, a *transcription* refers to a manual notation of a music performance and thus describes the transformation of an acoustic representation of a piece of music into a musical score or score-like representation. The transcription process is as old as the invention of musical notation systems. Those systems were specifically designed to describe performed music in such a way that it is possible to reproduce it at a different time or place. Notation systems are an attempt to capture the nature of music as an event-based

art form and to describe the properties of these events. Early notation systems restricted these properties to the sequential order of notes and rests, their duration and pitch. Modern notation systems additionally describe characteristics such as tempo, timbre, dynamics, and specific playing styles. Due to the diversity of the various musical concepts contained in a musical score, and the complexity of the extraction of this information from a recording, computational approaches to music transcription have usually restricted themselves to the extraction of individual note events, that is, pitch, note onset time and note offset time, without paying attention to other associated parameters such as metric position, relative note length or tonality.

Early AMT systems date back into the 1970s (Moorer, 1977), but research in this area has been increasingly carried out during the last 10-15 years. The reason for this considerable effort is the fact that music transcription is regarded as an enabling technology for a variety of music-related applications in different fields. Klapuri and Davy (2006) identify benefits for the following areas:

Music information retrieval: Retrieving music tracks based on melodic, harmonic or structural similarities to a given seed track.

Music processing: Modifying the acoustic content of a music track, such as the instrumentation or the arrangement. Decomposing a musical piece into its sources to enable separate processing and remixing.

Human computer interaction: Building accompaniment systems for live performances and systems that musically react to a performer. Supporting music students through performance analysis software.

Music-related equipment: Controlling music-related equipment such as light effects based on the musical content.

Musicological analysis: Analysing improvised music for which a notation does not exist and carrying out accurate analyses of music performances.

Transcription tools: Enhancing music notation software by offering a transcription of a given music recording.

Despite the comparably long history of research on music transcription algorithms, fully automatic music transcription systems are still error prone and often fail when more complex polyphonic music is analysed. The results of the *Multiple Fundamental Frequency Estimation & Tracking* task at the annual MIREX[1] evaluation indicate that the highest ranked algorithms are capable

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

of transcribing less than two-thirds of the played notes correctly in recordings of up to five instruments. This fact gives rise to the question of whether, and how, a human user could assist the computational transcription process in order to attain satisfactory transcription results. Certain skills possessed by human listeners, such as instrument identification and auditory stream segregation, are crucial for an accurate transcription of the musical content, but are often difficult to model algorithmically. Computers, on the other hand, are capable of performing tasks quickly, repeatedly and on large amounts of data. Combining human knowledge and perception with algorithmic approaches could thus lead to transcription results that are more accurate than fully-automatic transcriptions and that are obtained in a shorter amount of time than a human transcription. We refer to these approaches as *semi-automatic* or *user-assisted* transcription systems.

Involving the user in the transcription process entails that these systems are not applicable to the analysis of large music databases in the same way as fully-automatic transcription systems. Such systems can, however, be useful for detailed transcriptions of individual music pieces, and potential users could hence be musicologists, arrangers, composers and performing musicians. The main challenges of user-assisted transcription systems are to identify areas in which human input can be most beneficial for the transcription process, and to integrate high-level human knowledge into the low-level signal analysis. Different types of user information might thereby require different ways of incorporating that knowledge, which might include the application of user feedback loops in order to refine the estimation of individual low-level parameter estimates. Further challenges include the more practical aspects such as interface design and minimising the amount of information required of users.

This thesis focuses on the *algorithmic integration* of user input into computational transcription systems. Criteria for potential types of user input are identified and ways of utilising this information in a transcription framework are explored. Emphasis is placed on the investigation of information that enables the creation of accurate models for the instruments of the recording under analysis. The application of specific instrument models enables parts-based transcriptions, that is, the assignment of detected notes to the respective instruments, which has received little attention in the research history of AMT systems and which only recently has seen more interest.

Any empirical aspects of the user-assisted transcription process such as the analysis of inaccuracies in the user input or potential mistakes and their impact on the transcription results, practical investigations into the time and effort to provide the information, as well as any detailed considerations about the user interface, are out of scope of this thesis. Hence, no information from real human

users was harnessed in the development of the algorithms in this thesis. However, practical considerations were made regarding the criteria and constraints for the types of user information.

The target musical material consists of classical music pieces with a fixed and relatively small number of musical instruments. Recordings with multiple instruments are indispensable when parts-based transcription algorithms are to be evaluated. A fixed and limited number of instruments furthermore provides a controlled environment that enables detailed analyses. Music with percussive elements is explicitly excluded from the evaluations in this thesis.

1.2 Thesis structure

This thesis is structured as follows:

Chapter 2 reviews related work on AMT and summarises the most significant contributions in the field. Previous work that considers various forms of information from a human user is reviewed and criteria for potential types of user information are defined. Furthermore, the basic techniques that are used in the subsequent chapters are detailed: non-negative matrix factorisation (NMF), which can be used to factorise magnitude spectrograms into prototype spectra and activations, and the Viterbi algorithm for detecting the most likely state sequence of a Markov chain.

Chapter 3 investigates different ways of inferring timbre models for the instruments in a mixture based on information from a human user. Different types of user information and their corresponding timbre models are considered, and their suitability in a transcription setting is experimentally evaluated. Furthermore, a method is proposed to refine the timbre models based on the user-provided information.

Chapter 4 introduces methods that aim at limiting the amount of information a user has to provide in order to make the system more applicable in practice. Various ways of inferring missing information are considered and experimentally compared.

Chapter 5 proposes a pitch tracking method that tracks multiple instruments over time. The Viterbi algorithm is applied to find the most likely path through a number of candidate instrument and pitch combinations in each time frame.

Chapter 6 looks at instrument models that include phase information. The phase relations of harmonic partials are explored as a time-invariant phase property and the application for a user-assisted transcription task is demonstrated.

Chapter 7 finally concludes the thesis and provides future perspectives on user-assisted music transcription approaches.

1.3 Contributions

The following list contains the main contributions of this thesis and the chapters they appear in:

- Proposal of evalution metrics for the analysis of pitch activation functions in a multi-instrument context (Chapter 3).
- A comparison of generic timbre models and timbre models for the specific instruments in the recording (Chapter 3).
- Development of a method to extract multiple spectral templates from note annotations within a non-negative matrix factorisation framework (Chapter 3).
- Development and implementation of a source-filter model with a non-white excitation spectrum (Chapter 4).
- Development and implementation of a method to adapt generic sets of spectral templates to specific instruments of the same instrument type (Chapter 4).
- A comparison of various methods for the estimation of missing spectral templates (Chapter 4).
- Development of a multi-instrument pitch tracking based on a pitch activation function that takes into account the reconstruction error of the pitch-instrument combination, the pitch continuity as well as the instrument activity (Chapter 5).
- Exploration of instrument models that include the phase relations of harmonic partials as a step towards complex matrix decomposition (Chapter 6).

1.4 Associated publications

Most of the work in this thesis has been presented at international conferences or in journals. Additionally, several technical reports act as supplementary material for some of the conference publications.

Peer-Reviewed Conference Papers

[1] H. Kirchhoff, S. Dixon, and A. Klapuri. Shift-variant non-negative matrix deconvolution for music transcription. In *IEEE International Conference*

on Acoustics, Speech, and Signal Processing, pages 125–128, Kyoto, Japan, March 2012a

[2] H. Kirchhoff, S. Dixon, and A. Klapuri. Multi-template shift-variant non-negative matrix deconvolution for semi-automatic music transcription. In *13th International Society for Music Information Retrieval Conference*, pages 415–420, Porto, Portugal, October 2012c

[3] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri. Automatic music transcription: Breaking the glass ceiling. In *13th International Society for Music Information Retrieval Conference*, pages 379–384, Porto, Portugal, October 2012

[4] H. Kirchhoff, S. Dixon, and A. Klapuri. Missing template estimation for user-assisted music transcription. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 26–30, Vancouver, Canada, May 2013a

[5] H. Kirchhoff, S. Dixon, and A. Klapuri. Multiple instrument tracking based on reconstruction error, pitch continuity and instrument activity. In *10th International Symposium on Computer Music Multidisciplinary Research*, Marseille, France, October 2013b

[6] H. Kirchhoff, R. Badeau, and S. Dixon. Towards complex matrix decomposition of spectrograms based on the relative phase offsets of harmonic sounds. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Florence, Italy, 2014. submitted

Peer-Reviewed Journal Article

[7] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri. Automatic music transcription: challenges and future directions. *Journal of Intelligent Information Systems*, 41(3):407–434, 2013

Technical Reports

[8] H. Kirchhoff, S. Dixon, and A. Klapuri. Derivation of update equations for shift-variant non-negative matrix deconvolution (svNMD). Technical Report C4DM-TR-01-12, Queen Mary University of London, 2012b. `http://www.eecs.qmul.ac.uk/~holger/C4DM-TR-01-12`

[9] H. Kirchhoff, S. Dixon, and A. Klapuri. Derivation of update equations for multiple-template shift-variant non-negative matrix deconvolution based on β-divergence. Technical Report C4DM-TR-06-12, Queen Mary University

of London, 2012d. http://www.eecs.qmul.ac.uk/~holger/C4DM-TR-06-12

[10] H. Kirchhoff, S. Dixon, and A. Klapuri. Derivation of update equations for a source-filter model based on beta-divergence. Technical Report C4DM-TR-10-12, Queen Mary University of London, 2012e. http://www.eecs.qmul.ac.uk/~holger/C4DM-TR-10-12

[11] H. Kirchhoff, S. Dixon, and A. Klapuri. Cross-recording adaptation of musical instrument spectra. Technical Report C4DM-TR-11-12, Queen Mary University of London, 2012f. http://www.eecs.qmul.ac.uk/~holger/C4DM-TR-11-12

In publications [3] and [7] the author contributed sections on user-assisted music transcription as well as on instrument- and genre-specific transcription. For all other publications the author was the main contributor under the supervision of Dr. S. Dixon and Dr. A. Klapuri, and in the case of [6] Dr. R. Badeau.

Publications [1] and [2] are the basis for Chapter 3, publications [4] and [5] for Chapters 4 and 5, respectively, and submission [6] for Chapter 6. The derivations in Appendices A–C are based on the technical reports [9]–[11].

Chapter 2

Background

This chapter covers the background knowledge for this thesis. First, the basic terms that are used throughout this work are defined in Section 2.1. Section 2.2 provides an overview, a categorisation and a discussion of previous work on AMT as well as chonological overview of the most significant studies in tabular format. Based on the conclusions from prior AMT work, we motivate our study of user-assisted music transcription in Section 2.3. Prior work that utilises user-information for audio analysis tasks is reviewed and criteria for user input for transcription systems are defined. The remaining Sections 2.4 and 2.5 introduce techniques that are fundamental to the work presented in the remainder of this thesis: non-negative matrix factorisation (NMF) which is the basis for our analysis framework, as well as the Viterbi algorithm which is employed for the proposed pitch tracking method.

2.1 Terminology

Periodic signals

The starting point of all analyses in this thesis are digital recordings consisting of amplitude variations over time, which are denoted as *signals*. A signal is called *periodic* if its waveform continually repeats itself after a fixed amount of time. An illustration of a periodic signal is shown on the left hand side of Fig. 2.1. The time interval between two successive corresponding repeated values is denoted as the *fundamental period*. The *fundamental frequency (f0)* measures the number of fundamental periods per second and is given by the inverse of the fundamental period.

A periodic signal can be expressed by a sum of sinusoids with frequencies corresponding to integer multiples of the fundamental frequency. The periods of

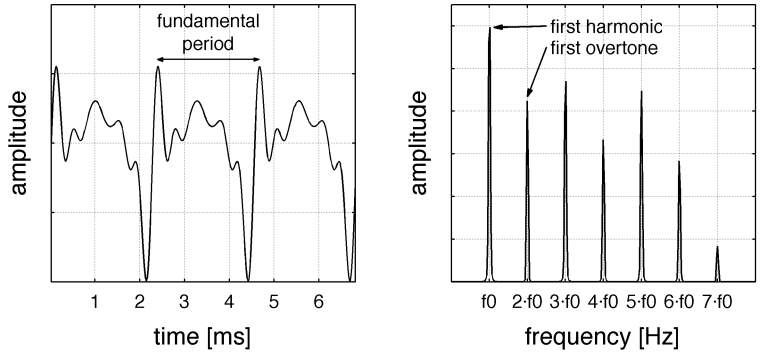


Figure 2.1: Illustration of a periodic signal. Left: periodic waveform in the time domain, right: magnitude spectrum.

these sinusoids correspond to integer divisions of the fundamental period. The shape of the periodic signal is determined by the magnitudes of the sinusoids as well as their phase angles. Figure 2.1 (right) displays the magnitude spectrum of the periodic signal. The individual peaks in this diagram correspond to the magnitudes of the sinusoidal components.

Musical instruments that produce periodic waveforms are denoted as *pitched instruments*. The sinusoidal components of these periodic sounds are called *harmonics* or *harmonic partials*. The fundamental frequency is equivalent to the *first harmonic*. An alternative terminology expresses the series of harmonics above the fundamental frequency as *overtones* (Federal Standard 1037C, 1996). The first overtone is equivalent to the second harmonic. These terms are illustrated in the diagram on the right hand side of Fig. 2.1.

A musical instrument that only plays one harmonic sound at a time is in this thesis denoted as a *monophonic* instrument as opposed to a *polyphonic* instrument which can produce several harmonic sounds simultaneously. The term 'monophonic' is used here differently from the monophonic playback format which denotes a single channel recording. Polyphonic also has a slightly different meaning in musicology where it is used to indicate several concurrent melodic lines.

Fundamentals of music notation

Music theory defines an *octave* as a fundamental frequency ratio of 2 between two harmonic sounds. In music with equal temperament an octave is divided into 12 equally spaced intervals, called *semitones*, which are denoted by a letter (C,D,E,F,G,A,B) and in some cases an additional *accidental* (♭, ♯, ♮). Each note in a musical score is defined by its name and its octave number (e. g. 'E♭4'

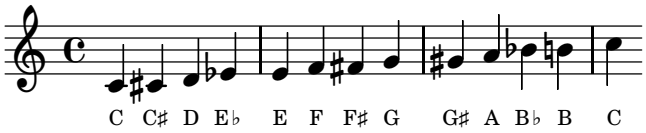
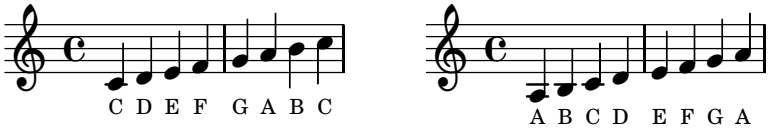


Figure 2.2: Notes of the chromatic scale.



(a) C major scale. (b) A natural minor scale.

Figure 2.3: Diatonic scales.

referring to the 'E♭' in the 4th octave). The ordering of all 12 semitones from low to high or high to low is called the *chromatic scale* and is illustrated in Fig. 2.2. The *diatonic major* and *minor* scales are composed of 7 notes with a fixed interval order and *root note* (see Fig. 2.3).

A *melody* describes a musically coherent sequence of notes, whereas a *chord* denotes simultaneously sounding notes that form a *harmony*. The *key* of a musical piece defines its tonal centre (e. g. 'B♭') as well as the main scale (e. g. 'minor'). Notes, melodies and chords can be played at various *dynamic levels*. In scores of Western classical music these are denoted by symbols such as ***p*** (piano), ***mf*** (mezzoforte) or ***f*** (forte). Dynamics determine the volume of a sound which usually result in different perceived *loudness levels*. The relation between the energy of a sound and the perceived loudness is non-trivial (Zwicker and Fastl, 1999, Chapter 8).

Pitch

The *pitch* is a perceptual attribute of a sound and describes its perceived tone height. An ANSI standard (1994) defines it as "that attribute of auditory sensation in terms of which sounds may be ordered on a scale extending from low to high". Hartmann (1996) provides another definition that measures pitch by "adjusting the frequency of a sine wave of arbitrary amplitude". In this thesis we will assume that the perceived pitch is determined by the fundamental frequency of a harmonic sound and we will use the terms *pitch* and *fundamental frequency* interchangeably.

A pitch of a *note* in a musical score is here denoted as its *nominal pitch*. The nominal pitch can either be defined by its note name and octave (see above)

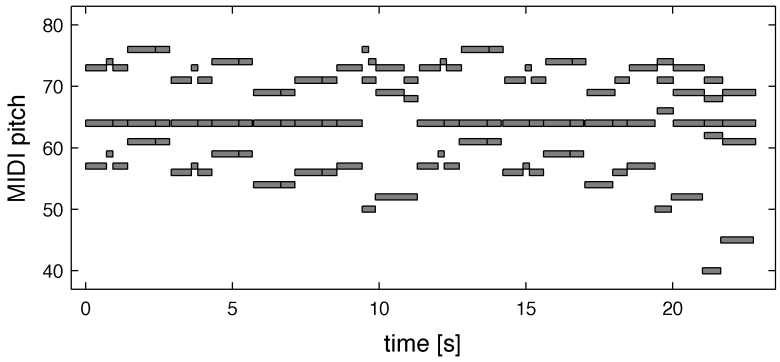


Figure 2.4: Piano roll representation of the beginning of W. A. Mozart's Sonata No. 11 in A major (KV331).

or by its *MIDI pitch*. The MIDI pitch is a value that enumerates notes on a chromatic scale from C0 to G10 resulting in 128 different note numbers.

Even though a note in the score has a single nominal pitch, its fundamental frequency in a performance can exhibit variations over time. *Vibrato* describes a deliberate periodic modulation of the pitch, whereas a *glissando* is a continuous pitch change, usually over a wider pitch range. Different *tuning* and *temperament systems* cause fundamental frequencies to deviate from the equal tempered scale, which defines the fundamental frequency ratio between adjacent semitones as $2^{\frac{1}{12}} \approx 1.059$.

Piano roll

Traditionally, a piano roll was used to control self-playing pianos, called player pianos, which were popular in the late 19th and the early 20th centuries (Bowers, 1972). These mechanically controlled pianos enabled the recording and reproduction of piano performances and at the same time provided the opportunity to retrospectively edit a recording even before the invention of electronic recording devices. A piano roll was essentially a paper roll with rectangular holes, each corresponding to an activated key on the piano. The length of the hole controlled the duration the key was pressed and hence specified the length of the note. The vertical position of the holes determined their nominal pitch and the horizontal position their temporal location.

The analogy of a piano roll is often used in research on AMT systems as it represents information extracted from a musical performance in a format that is similar to a musical score. A piano roll displays the *onset* and *offset times* of notes as well as their nominal pitch. Different *instrument parts* can simply be displayed by separate piano rolls. A piano roll representation is often also used

to display the performance data stored in a *MIDI file* and it is often used in standard audio and MIDI software. An example of a piano roll representation of the beginning of W. A. Mozart's Sonata No. 11 in A major (KV331) is displayed in Fig. 2.4.

2.2 Research on automatic music transcription

Research on automatic music transcription has been carried out for almost 4 decades. The early work of Moorer (1975) is seen as the first noteworthy approach to computational polyphonic music transcription. Since then, numerous studies have been published — the vast majority within the last 10-15 years. The large number of publications makes it impossible to give an exhaustive overview of the development of transcription systems. We aim here at discussing the most important contributions only.

In the literature, different terms can be found to describe automatic music transcription systems. Although these terms are not always used consistently, certain conventions have emerged: The term *f0 estimation* is generally used to describe systems that detect fundamental frequencies (f0s) within each individual analysis frame of the audio spectrogram. Depending on the target audio material, these systems are either *single f0 estimation* or *multiple f0 estimation* systems. Multiple f0 estimation systems can be further divided into *single instrument* and *multiple instrument* multiple f0 estimation systems. *Predominant f0 estimation* aims at extracting the most prominent pitch in each time frame and is often used as a synonym for *melody extraction* even though the predominant pitches might not always be part of the same melody. In order to combine the frame-based estimates into note objects, a post-processing step is necessary which is denoted as *note tracking*. *F0 tracking* and *pitch tracking* are used to describe the process of combining frame-based pitch estimates into melodic streams. The terms *polyphonic pitch estimation* or *multipitch estimation* only refer to the fact that pitches are estimated from a polyphonic audio signal. Thus, these terms can refer to both frame-based and note-based systems. However, there seems to be a tendency to use these terms more often for note transcription systems.

A categorisation of existing AMT methods is not straightforward since transcription systems can be quite diverse and usually consist of multiple processing stages that do not necessarily follow a uniform processing chain. Therefore, several taxonomies have been proposed to characterise polyphonic transcription systems in previous work. The overview work by de Cheveigné (2006) categorises methods based on their signal representation into *temporal*, *spectral* and *spectrotemporal* approaches. Although this classification is complete, it is not very meaningful since the signal representation is only the first step in a number of

processing steps and does not reveal anything about how the individual notes are detected. Yeh (2008) classifies AMT systems into *iterative* and *joint* estimation systems. Iterative estimation techniques repeat the process of estimating the spectrum of the most prominent fundamental frequency and subtracting it from the spectrum of the musical mixture until a certain termination condition is met, whereas joint estimation techniques aim at finding the most likely *combination* of fundamental frequencies. The literature reviews by Benetos (2012) and Klapuri (2004b) organise the work based on the main underlying principles. Although this type of categorisation might not be complete and can even be ambiguous for some publications, it identifies the main computational techniques and enables a comparison between them. This type of categorisation will be adopted for the review in this work.

The following Sections 2.2.1–2.2.6 will introduce the different categories and detail several approaches. In Section 2.2.7, a chronological overview of the most significant work is provided. The different categories are discussed in Section 2.2.8 where we will also give an overview of the most successful approaches of the annual MIREX evaluations.

2.2.1 Perceptually-motivated approaches

Perceptually motivated AMT systems apply either psychoacoustical knowledge or models of the physiology of the human auditory system to the analysis of sound mixtures. Most of these systems were proposed in the 1990s. *Psychoacoustical knowledge* refers to findings about how the human brain makes sense of an auditory scene which is described as *auditory scene analysis (ASA)*. Bregman (1994) was one of the main contributors to ASA. The algorithmic formulation of these research results is denoted as *computational auditory scene analysis (CASA)*. Important contributions to CASA were made by Ellis (1996). Knowledge about the *physiology* of the human auditory system uses findings about the different sound processing stages of the human auditory system, i. e. the outer, middle and inner ear (Moore, 1995).

Computational auditory scene analysis (CASA)

Kashino and Tanaka (1993) were among the first to apply Bregman's findings to the problem of source separation and transcription. The authors employ sinusoidal modelling techniques in order to find partial tracks. The ASA cues of *harmonicity* and *onset synchrony* are applied in order to group partial tracks into note candidates. Several acoustic features are used to cluster the note candidates into sound sources. In order to resolve overlapping partials, pre-trained *tone*

models are used which are based on the ASA concept of *timbre memory* and an *old-plus-new-heuristic* of sound perception.

Sterian (1999) in his Ph. D. thesis employed even more of Bregman's grouping principles in order to cluster sinusoidal partials which were extracted by a Kalman-filter-based approach. For each of the principles *common onset and offset time*, *harmonicity*, *low partial support (strong lower partials)*, *lack of partial gaps*, and *partial density* a likelihood function is computed and all these functions are combined to compute a likelihood measure for a given combination of partial tracks. The amount of partial combination hypotheses is limited by a pruning technique.

Godsmark and Brown (1999) denote their system as a *blackboard architecture* that sequentially applies physiological and psychacoustic processes of the human auditory system to the audio signal. The physiological part extracts so-called *synchrony strands* from the signal which are similar to sinusoidal tracks. The psychoacoustic part groups these strands into sound sources. Several independent expert processes based on CASA principles are combined to form the most likely organisation hypothesis for the synchrony strands.

Physiological modelling

Several approaches have been proposed that are based on the *unitary model of pitch perception* for individual pitches by Meddis et al. (Meddis and Hewitt, 1991; Meddis and O'Mard, 1997). In this model, the outer and middle ear transfer functions are approximated by a second-order band-pass filter that attenuates the amplitudes of high and low frequency components of the input signal. 60 gammatone band-pass filters on a logarithmic frequency scale model the behaviour of the basilar membrane. A specific model for the inner hair cells of the basilar membrane is employed and the periodicities in each channel are computed by the short-time autocorrelation function (ACF). All ACFs of the different filter bands are then summed, which is denoted as the *summary autocorrelation function (SACF)*, and the maximum of this sum indicates the perceived pitch.

De Cheveigné and Kawahara (1999) built the unitary pitch model into a multiple-f0 detection system. The estimation is performed either jointly or by iterative cancellation. The latter iteratively estimates periodicities based on the peaks in the ACF and removes all channels that contain peaks at the same position. The joint estimation uses a concatenation of several fixed-period filters and exhaustively adjusts their parameters until the input signal is suppressed.

The system by Martin (1996a,b) employs the output of several gammatone filters to compute a logarithmic lag correlogram in the same way as proposed by

Ellis (1996). The correlogram is further processed by a blackboard system that detects notes based on the peaks in the SACF.

The approach by Tolonen and Karjalainen (2000) splits the input signal only into two auditory channels above and below 1 kHz. Half-wave rectification is applied to the higher band only. After periodicity detection, the SACF is enhanced by a peak pruning technique in order to detect the periodicities contained in the mixture. The method was evaluated only qualitatively on synthetic sinusoids as well as a combination of vowels and 2–4 voice clarinet chords.

Klapuri (2004a, 2005) proposed a multiple-f0 detection system that applies several modifications to the unitary pitch model. The period estimation by the ACF is replaced by a technique called *harmonic selection* that takes only certain harmonics into account and a more complex weighting technique is employed when summing the subbands. The system was evaluated on sample-based random mixtures of recorded instruments.

2.2.2 Heuristic methods

A considerable number of approaches apply heuristic rules to extract fundamental frequencies from the time-frequency representation. These rules often take into account knowledge about acoustical properties including temporal and spectral characteristics of harmonic sound sources, the ways in which these sources interfere, and the contributions of the acoustic environment in which they were recorded.

The work by Moorer (1975, 1977) can be seen as the first attempt to approach the problem of automatic music transcription. The target material was restricted to be performed without vibrato and glissando, the fundamental of a note was not supposed to overlap with a harmonic of another note and instrument voices were not allowed to cross each other. The method passes the audio through a bank of equally-spaced filters, the distance of which is determined adaptively by an initial periodicity analysis of the audio mixture. The power and frequency of each filter output is then analysed and a numerical score is computed that characterises the strength of each harmonic. Based on these scores, note hypotheses are formed and notes are detected by discarding unlikely hypotheses. Notes with considerable time-overlap are assigned to the different sound sources and the remaining notes are heuristically assigned to one of the instruments. An attempt was also made to produce a printable score sheet.

Another early approach by Maher (1990) aimed at transcribing and separating duet signals. The approach uses sinusoidal modelling techniques to detect the partial trajectories of the recording. In order to detect the fundamental

frequencies, the so-called *two-way-mismatch (TWM)* procedure is employed which measures the mismatch between the predicted partial sequence of an f0 candidate and the observed partials within a frame. The two voices are jointly estimated within user-specified ranges by repetitively keeping one voice fixed while the best match is made for the other voice.

A pattern can be recognised in the processing stages of various heuristic approaches: after computing some form of time-frequency representation and applying a noise threshold, a peak-picking stage picks the most salient spectral components. These are usually further processed to find pitch candidates or a pitch salience function by applying heuristic rules. Candidates or saliences are subsequently employed to track notes over time.

The early polyphonic pitch detection method by Chafe et al. (1985) picks the peaks in each frame of a bounded-Q spectrogram and combines them in order to determine f0 candidates. Based on these candidates, notes are tracked in forward and backward directions and spurious notes are eliminated. Once all notes have been determined, the corresponding peaks are cancelled from the spectrogram and the search is repeated for the residual peaks.

Klapuri (2003) proposed an iterative method for detecting fundamental frequencies in an audio frame. After an initial noise reduction stage the algorithm estimates the predominant f0 based on a subband analysis of the spectral content which allows partials to slightly deviate from their exact harmonic position. In order to account for overlapping partials between different sources, spectral smoothing is applied to estimate the amplitude of the partials of the predominant f0 before subtracting the resulting sound spectrum from the mixture. A polyphony estimation procedure is applied to stop the iterative f0 estimation when no more harmonic sources are found in the residual spectrum.

Another transcription system by Klapuri (2006) uses the sum of the harmonic partial amplitudes as an f0 salience function. Spectral whitening is performed before the computation of the salience function and an exponentially decaying weighting function was found most useful for the partial summation process. Both iterative and joint f0 estimation techniques were evaluated and a fast implementation was proposed for the iterative estimation procedure. While the results of iterative and joint estimation procedures were comparable for lower polyphonies, the joint estimation achieved better results for high polyphonies.

Pertusa and Iñesta (2008) use a fixed noise threshold in order to discard low energy peaks in each frame of the STFT. A fixed number of f0 candidates is selected by summing the harmonic amplitudes and choosing those with the largest values. Instead of using a polyphony estimation stage, all combinations of f0 candidates are considered and a salience function for each combination is computed based on the loudness and the spectral smoothness of each candidate.

Interpolation between non-overlapping partials is used to estimate the amplitudes of overlapping partials.

Cañadas Quesada et al. (2010) apply noise thresholding and peak-picking to the magnitude spectrogram and identify f0 candidates in each time frame. All possible candidate combinations of up to five concurrent pitches are modelled by a sum of Gaussians and the distance measure based on the Euclidean distance is computed that also compares the Gaussian approximation to the spectra at previous time frames. A two-state HMM is used for note tracking.

The algorithm of Yeh et al. (2010) uses a more sophisticated noise reduction stage that adaptively computes the noise envelope within each frame based on a Rayleigh distribution. Only spectral peaks above the noise threshold are considered for the subsequent f0 estimation. The joint estimation stage evaluates hypothetical fundamental frequencies based on a score function that combines four different criteria: the harmonicity, mean bandwidth, spectral centroid and the synchronicity of the partials. A candidate selection method extracts a set of harmonically unrelated f0s and subsequently looks for harmonically related f0s. A polyphony inference algorithm is used to estimate the number of sources within each frame.

The approach by Argenti et al. (2011) is based on a so-called bispectral analysis front-end. The bispectrum resembles the covariance matrix of the CQT analysis spectrum at each time frame. Pitches are detected by correlating each bispectrum with a 2-dimensional bispectral pattern for each possible pitch. Note candidates are iteratively cancelled from the original CQT spectrum by applying the spectral smoothness principle. Note durations are independently extracted from the spectrogram and are matched with the pitch candidates.

Dressler's system (2011; 2012) was originally designed for melody extraction but achieved very good results in the *Multiple F0-Estimation & Tracking* task at the MIREX evaluations in 2011 and 2012. It relies on a pairwise evaluation of spectral peaks at each time frame of a weighted multi-resolution magnitude spectrogram. Peak pairs are considered as harmonics with successive harmonic indices or successive *odd* harmonic indices of a virtual pitch. Further criteria such as harmonicity strength, spectral smoothness, presence of intermediate peaks and harmonic number influence the overall pitch salience. Notes are tracked heuristically and spectral envelopes are extracted for each note.

The recent system by Grosche et al. (2012) computes a multi-resolution spectrogram, applies instantaneous frequency estimation and noise suppression and converts it to a semitone spectrogram. Pitch saliences are computed from the semitone spectrogram in a similar fashion as in Klapuri (2006) and combined with an onset detection stage. HMMs are employed for the note tracking stage.

2.2.3 Approaches based on probabilistic inference

Probabilistic approaches to music transcription model the problem by means of a statistical framework. The framework typically formulates certain general assumptions about the sound sources and the mixture, and estimates the model parameters by statistical inference. These methods estimate the likelihoods of certain sets of fundamental frequencies within each analysis frame and select the most likely combination.

The *PreFEst* algorithm by Goto (2004) separately transcribes the melody and bass lines of a music signal. It splits the spectrum of the signal into a high-pass filtered and a low-pass filtered part and detects the predominant f0s in each of these spectra. A probabilistic tone model is employed which describes a tone as a number of equally spaced harmonics and models the height and exact position of each harmonic by a Gaussian distribution. The expectation-maximisation (EM) algorithm is used for the estimation of both the weight of each tone model and its shape, i. e. the amplitudes of the harmonics. The results of the EM procedure are postprocessed by a multi-agent architecture to ensure temporal continuity in the pitch salience function.

Goto's PreFEst algorithm was extended by the *harmonic temporal structured clustering (HTC)* algorithm of Kameoka et al. (2007). Similar to the PreFEst algorithm, it models harmonic spectra as a sum of Gaussian distributions at integer multiples of the fundamental frequency. Additionally, the time evolution of each partial is likewise modeled as a sum of Gaussian distributions that are equally spaced in time and the partials are grouped into clusters. The EM algorithm is used to estimate the parameters for each source model and to decompose the spectrogram.

Another extension of Goto's PreFEst algorithm was proposed by Yoshii and Goto (2012). The authors propose a system based on infinite latent harmonic allocation (iLHA) which has the potential to estimate the complexity, that is, the number of simultaneous F0s along with the other model parameters. Probabilistic inference is carried out by a variation of the variational Bayes (VB) method.

The system of Ryynänen and Klapuri (2005) is built on the multiple-f0 system by Klapuri (2004a, 2005). It applies three probabilistic models to detect notes in the recording: a note-model, a silence model and a musicological model. The note model uses a three-state HMM to represent temporal segments. The musicological model determines the transition probabilities between the note HMMs. These are learned from a database of monophonic melodies. A *token-passing algorithm* is used to compute the most likely path through the different probabilistic models.

Davy et al. (2006) propose a statistical framework that models harmonic sounds by a superposition of Gabor atoms. The atoms are estimated on an equidistant time axis but the frequencies depend on the positions of the harmonic partials of the instrument sources. Parameters are estimated by the maximum-a-posteriori (MAP) method, and a hierarchical prior structure is employed. In order to evaluate integrals in the parameter estimation stage, the Markov-Chain-Monte-Carlo (MCMC) method is used.

An approach by Emiya et al. (2010) models the spectral envelope of piano sounds by an autoregressive (AR) process and employs a moving average (MA) noise model. A heuristic preprocessing step is applied to find the most salient pitch candidates. Parameters of the note model and the noise model are iteratively estimated to find the most likely combination of pitches. An additional major contribution of this work is the creation of a groundtruth database of MIDI aligned piano sounds (MAPS) for piano transcription.

Another piano transcription model was proposed by Raczynski et al. (2010). The authors employ a dynamic Bayesian network (DBN) to model the conditional probability of chords, note activations, and their relation to observed pitch saliences. Saliences are extracted by non-negative matrix factorisation and the conditional probabilities are estimated directly from the MIDI ground truth of the dataset.

2.2.4 Spectrogram factorisation methods

Non-negative matrix factorisation

Another group of methods that has attracted attention in recent years is based on a technique called *non-negative matrix factorisation (NMF)*. NMF was introduced in a seminal paper by Lee and Seung (2001) and aims at decomposing a time-frequency representation into a matrix that contains the spectra of the individual sounds and another matrix that contains the information about when each of these spectra is active. An in-depth introduction to NMF will be given in Section 2.4.

Smaragdis and Brown (2003) were the first to apply NMF to music analysis. NMF is used to factorise the STFT magnitude spectrogram of a piano music excerpt and various examples of isolated and coinciding sounds were presented.

Bertin et al. (2007) compared NMF to non-negative K-SVD, another matrix factorisation technique. After factorising the spectrogram, the pitch of each atom is determined and the activations are thresholded to determine onset and offset times. The authors investigated the influence of varying numbers of spectral basis functions and the effect of initialisation on the overall transcription results

of solo piano music. There was no major benefit from the initialisation but a slight increase in performance with larger numbers of basis functions.

Cont (2006) used NMF for real-time multipitch analysis. As opposed to Samragdis' and Bertin's methods above, this method does not apply NMF in a completely blind way to the analysis spectrogram. Spectral basis functions for the instruments in the recordings are learned off-line in advance, and are kept fixed in the analysis stage. Incoming STFT analysis spectra are decomposed in real-time by determining the activations of the pre-learned spectra. The authors also introduce a sparsity constraint on the activations based on the hyperbolic tangent. This approach was extended in 2007 and the modulation spectrum was employed as the spectral representation (Cont et al., 2007).

Pre-learned spectra were also used by Niedermayer (2008). The basis functions – here denoted as tone models – are learned from a database of individual notes of the instrument type under analysis. Activations were smoothed by a median filter and thresholding was applied to detect note events.

Raczyński et al. (2007) enforce harmonicity on the basis functions by setting the basis functions at the frequency bins between harmonic partials to zero. Due to the multiplicative update rules of NMF, these frequency bins will remain zero when the basis functions are updated. A further constraint is targeted towards reducing the correlation between the activations in order to reduce octave errors.

A different approach towards harmonic basis functions was proposed by Vincent et al. (2008). Basis functions are modelled by a weighted sum of narrowband spectra each of which contains a small number of harmonic partials. With this approach different spectral envelopes can be modelled. Slight inharmonicities and tuning differences were also taken into account.

The issue of modelling varying instrument timbres was also addressed by Grindlay and Ellis (2011). NMF is used as a rank reduction technique for different timbres. Spectra of each instrument are concatenated into a vector, and the matrix containing all instrument spectra is factorised into so-called *eigeninstruments*. Another hierarchy level is introduced that computes the eigeninstruments individually for several instrument families.

Hennequin et al. (2010) proposed a method to overcome the static nature of the basis functions. The authors employ a parametric model for the basis functions that enables the detection of time-varying fundamental frequencies. A set of partial amplitudes is learned for each pitch of the chromatic scale. Sparsity constraints, activity uncorrelation constraints and spectral smoothness constraints are additionally employed.

The method by Hennequin et al. can be described as *scale-invariant* because it combines spectral templates with varying fundamental frequencies which results in a scaling of the frequency distances between the harmonic partials.

When a time-frequency analysis with a *logarithmic frequency axis* is used, varying the fundamental frequency results in a *translation* of the spectrum along the frequency axis. This property is denoted as *shift-invariance*. Shift-invariant NMF was proposed by FitzGerald et al. (2005) as *shifted NMF*. The algorithm allows a single template to be used in a restricted fundamental frequency region (cf. Section 2.4.3.2).

As an extension to NMF, Smaragdis (2004) proposed a technique called *non-negative matrix factor deconvolution* that allows the basis functions to have a temporal extension, as opposed to single frame templates. Hence, spectrogram fragments can be used as basis functions. This method is described in greater detail in Section 2.4.3.1.

The concepts of shift-invariance and time-extended basis functions were combined by Schmidt and Mørup (2006b). The method was named *non-negative matrix factor 2-D deconvolution (NMF2D)* and enable frequency shifts of time-extendend basis functions (cf. Section 2.4.3.3).

Probabilistic latent component analysis

Probabilistic latent component analysis (PLCA) is another matrix factorisation technique. It interprets the spectrogram as a 2-dimensional probability distribution and factorises it into a product of latent marginal distributions. It can be shown (Shashanka, 2008) that PLCA is equivalent to non-negative matrix factorisation. Its mathematical formulation is given by (Shashanka, 2008)

$$P(f,t) = \sum_z P(z)P(f|z)P(t|z). \tag{2.1}$$

In this equation, $P(f,t)$ represents the probabilistic interpretation of the magnitude spectrogram. $P(f|z)$ denotes the z-th basis function, and $P(t|z)$ the corresponding time activations. Both terms $P(f|z)$ and $P(t|z)$ can be represented by a matrix where one dimension represents the latent variable index z and the other dimension the discrete frequency and time indices, respectively. $P(z)$ is a scaling factor that compensates the fact that the marginal distributions sum to 1.

Since PLCA is just a different formulation of NMF, some of the above mentioned NMF extensions have also been formulated in a probabilistic way. The probabilistic formulation makes PLCA in general attractive for probabilistic extensions and Bayesian modelling.

The basic PLCA model was formulated by Smaragdis et al. (2006) as an extension to probabilistic latent semantic indexing (PLSI) (Hofmann, 1999). PLSI is a dimensionality reduction technique that has its origins in the field

of text-based information retrieval. PLCA was introduced as a general sound analysis technique for feature extraction, sound recognition, sound separation and denoising.

A shift-invariant extension of the basic PLCA model was introduced by Smaragdis and Raj (2007). It enables shifts in both the horizontal and the vertical direction of single-frame and time-extended basis functions. The authors demonstrated the use of the method not only for music and speech analysis but also for image processing. This method can be seen as the equivalent to Schmidt and Mørup's NMF2D model described above.

Mysore and Smaragdis (2009) used shift-invariant PLCA for note transcription. A Dirichlet prior is employed to promote unimodal activity distributions and a Kalman filtering based temporal continuity constraint is used in order to discourage activities from varying between different basis functions.

Fuentes et al. (2011) incorporate the timbre modelling technique proposed by Vincent et al. (2010) into shift-invariant PLCA. Instrument spectra are modelled by a weighted sum of narrowband spectra and can be translated along the logarithmic frequency axis. Additionally a prior was introduced in order to minimise octave confusions. Although a polyphonic model was presented, the method was tested on isolated monophonic notes.

An attempt to overcome the static nature of the basis functions was made by Mysore et al. (2010). The authors incorporate a non-negative Hidden Markov model (N-HMM) into the PLCA model which enables capturing the temporal structure of sounds. Prior to analysis, N-HMMs are learned for all expected source types in the mixture and a factorial HMM (N-FHMM) combines the individual N-HMMs. The model was applied for speech separation. This approach was extended for music transcription by Benetos et al. (2013). The basic PLCA model is replaced by a shift-invariant PLCA model with the ability to capture frequency modulations and tuning variations. The states of the HMMs are expected to capture the attack, sustain and decay states of a note and are learned for each pitch of each source individually from isolated note templates. These temporally-constrained note models are subsequently employed for transcription and sparsity constraints are applied on the note activations.

Hoffman et al. (2010) address the problem of automatically estimating the number of latent components which is a fundamental issue with latent variable models. Their Gamma Process Non-negative Matrix Factorisation model (GaP-NMF) applies nonparametric Bayesian methods to simultaneously estimate the model complexity (i. e. the number of latent components) as well as the basis functions themselves. In this study, the ability of GaP-NMF to correctly estimate the number of components was evaluated in comparison with other NMF variants. No results were provided for the transcription accuracy of the algorithm.

Specmurt

The term *specmurt*, an anagram of 'spectrum', was introduced by Saito et al. (2008) and refers to the inverse Fourier transform of a spectrogram with log-scaled frequency. The algorithm estimates a *common harmonic structure*, a common spectral template, by iteratively deconvolving the spectrum with the template, de-emphasising small peaks and re-estimating the spectral template. Even though the specmurt algorithm is not strictly a matrix factorisation technique, the underlying ideas are comparable to some of the matrix factorisation techniques described above: The deconvolution of the log frequency spectrum with a common harmonic structure is very similar to shift-invariant NMF and PLCA, and the attenuation of small peaks is aimed at removing spurious activations and making the results more sparse.

2.2.5 Methods based on sparse coding

In the same way as spectrogram factorisation techniques, *sparse representations* approximate the spectrogram by a superposition of single frame prototype spectra. The prototype spectra are often denoted as *spectral atoms* and the set of prototype spectra is called a *dictionary*. The specific characteristic of sparse representations is the assumption that only a fraction of atoms is active at any given time. This can be expressed by a minimisation of the l_0-norm of the activities at each time frame. Sparse approximations usually employ overcomplete dictionaries, that is, dictionaries in which the number of spectral atoms exceeds the dimensionality of the data (Olshausen and Field, 1997).

Since the l_0-norm is non-convex, a cost-function between the original magnitude spectrogram and the sparse approximation cannot be optimised by conventional gradient descent methods. The sparse approximation problem is usually approached either by greedy methods such as *Matching Pursuit (MP)* and *Orthogonal Matching Pursuit (OMP)*, or by convex optimisation. Matching pursuit (Mallat and Zhang, 1993) selects the dictionary element that exhibits the largest inner product with the analysis spectrum, subtracts its contribution from the analysis spectrum, and removes it from the dictionary. This process is repeated for the remaining dictionary elements until a stopping criterion is met. This method is very similar to iterative estimation and cancellation method by Klapuri (2006). Orthogonal Matching Pursuit omits the step of subtracting dictionary elements from the analysis spectrum and instead takes the already selected atoms into account when computing the inner product of the remaining dictionary elements. *Basis pursuit (BP)* (Chen et al., 1998) is a convex optimisation technique that replaces the l_0-norm of the activations by the l_1-norm. The cost function thus consists of the reconstruction error between the

original spectrogram and the sparse approximation, plus the weighted l_1-norm. The weighting factor provides control over the sparsity of the solution.

Abdallah and Plumbley (2004) use an ML approach to learn the dictionary from the power spectrogram, and MAP estimation to infer the activations in a probabilistic model based on sparsity. The sparsity is enforced by prior distributions on the activities that are strongly peaked around zero, thereby assigning a high probability to low activations and a low probability to larger activity values. The algorithm was evaluated on a single piece of piano music and included some manual processing steps and a visual examination of the transcription result. Based on this rather informal evaluation, extraordinarily high transcription accuracies were reported.

Blumensath and Davies (2004) learn atoms in the time domain by an *iterative reweighted least squares (IRLS)* procedure. At each time frame, a subspace is selected by a variant of OMP. Given the subspace, the activations are computed taking into account a log-l_1-norm constraint, and the dictionary matrix is updated.

The system by Cañadas Quesada et al. (2008) is based on the *harmonic matching pursuit (HMP)* method. HMP employs harmonic basis functions defined by their fundamental frequency and the amplitudes of the harmonic partials. As in MP, the best harmonic atom in each iteration is removed from the analysis spectrum and the process is repeated. An additional spectral smoothness constraint is applied to resolve the partials of pairs of notes with integer frequency relations.

Leveau et al. (2008) likewise employ instrument-specific harmonic atoms. A variant of the MP algorithm is employed that adjusts the fundamental frequency and the phases of the harmonic partials before subtracting the atom from the signal. Successive atoms are organised into so-called molecules. The algorithm is not only applied for music transcription and visualisation but also for monophonic and polyphonic instrument recognition.

Lee et al. (2011) combine heuristic methods and sparse coding for piano transcription. For each piano note, several templates are estimated in advance. A heuristic note candidate selection is performed for each frame individually, and a dictionary is built for each time frame containing all atoms of the note candidates. BP is employed to compute a sparse solution and the sparse coefficients are temporally smoothed by an HMM.

O'Hanlon et al. (2012) employ the concept of *group sparsity* (or *structured sparsity*) for piano transciption. Group sparsity assumes that spectral atoms are organised in groups and applies the sparsity constraint only to the groups rather than the whole set of atoms. No sparsity is imposed on the atoms within each

group. The atoms for each group are learnt in advance from a piano database and several variants of the algorithm were applied.

2.2.6 Classification-based approaches

A number of transcription algorithms employ classification algorithms from the field of machine learning. In this context, classifiers are trained for each individual pitch that is to be recognised in a *one-vs-all* approach. The output of these classifiers indicates the likelihood of each pitch and therefore acts as a pitch activation function.

Marolt (2004) was among the first to apply a classification algorithm — in this case *neural networks* — to music transcription. Several neural network topologies such as multilayer perceptrons (MLP), radial basis functions (RBF) and time-delay neural networks (TDNN) were experimentally compared for the transcription of piano music. The author reports that TDNNs with three consecutive 10 ms time frames achieved the best classification results. In a similar way, Pertusa and Iñesta (2005) evaluated TDNNs for music with timbres other than the piano. The authors highlight the sensitivity of TDNNs to variations in timbre: the transcription accuracy considerably decreased when a network was trained on one timbre and tested on another. A different kind of neural network, a bidirectional recurrent neural network, was used by Böck and Schedl (2012). The authors use two STFTs with different time-frequency resolutions and reduce both to a semitone resolution. The spectrograms and their first differences act as inputs to the neural network. The system was trained and tested on solo piano music.

In addition to neural networks, *support vector machines (SVMs)* have also been used for the classification of audio frames. Poliner and Ellis (2007a,b) used SVMs with linear and RBF kernels for the classification of piano spectra. 88 SVMs were trained as one-vs-all classifiers on spectral features derived from the STFT analysis spectrum. The system by Costantini et al. (2009) takes a very similar approach with RBF kernel SVMs. A preliminary onset detection stage is employed and the subsequent CQT analysis specifically targets the signal parts following detected onsets. The CQT spectra are used as input features for the note classification SVMs and an offset detection procedure is proposed. Zhou (2006) likewise used RBF kernel SVMs. In this method, only the peaks of the time-frequency representation were used and all other frequency bins were set to zero.

The recent paper by Nam et al. (2011) employs Deep Belief Networks for unsupervised feature learning from the spectrogram and a subsequent SVM with a linear kernel for note classification.

2.2.7 Chronological overview

The following table lists the publications reviewed in Sections 2.2.1–2.2.6 in chronological order. Besides authors and year, the table includes for each publication the employed signal representation, the main processing stages, as well as information about the audio signals the algorithm was evaluated on.

Year	Author	Signal representation	Processing stages	F0 estimation	Polyphony	Evaluation	Category
1977	Moorer	filter-bank analysis	score function for each filter output, f0-estimation based on note hypotheses	iterative	2	synth. violin duet	heuristic
1985	Chafe et al.	bounded-Q	peak picking, frame-based partial grouping, note tracking	iterative	n/a	n/a	heuristic
1990	Maher	STFT	sinusoidal modeling (peak picking & partial tracking), two-way-mismatch procedure	joint	2	synthesised and acoustic examples	heuristic
1993	Kashino and Tanaka	filter-bank analysis	sinusoidal modeling (peak picking & partial tracking), partial grouping by auditory cues, sound clustering based on acoustic features, resolution of overlapping partials by tone models	joint	2–3	MIDI-generated chords	perceptually-motivated
1996	Martin	auditory filterbank	log-lag correlogram, blackboard system to detect notes	joint	4	synthesised 4-voice Bach chorales	perceptually-motivated
1999	Sterian	modal distribution	Kalman-filter based partial tracking, likelihood computation of partial hypotheses based on ASA cues	joint	1–4	synthesised and recorded brass instruments	perceptually-motivated
	De Cheveigné and Kawahara	auditory filterbank	unitary pitch model followed by period estimation within auditory channels	joint & iterative	2	synth. waveforms	perceptually-motivated
	Godsmark and Brown	auditory filterbank	formation of synchrony strands, blackboard architecture to group strands into sound sources	joint	2-7	3 sythesised MIDI excerpts	perceptually-motivated
2000	Tolonen and Karjalainen	LP and HP filter	Reduced unitary pitch model and enhanced SACF.	joint	2–4	synth. sinusoids, vowels and clarinet chords	perceptually-motivated
2003	Klapuri	STFT	noise suppression, predominant-f0 estimation, spectral smoothing, iterative subtraction of spectra	iterative	1–6	random and musical mixtures	heuristic
2004	Goto	STFT / instant. freq.	LP & HP filtering the spectrum, probabilistic estimation of f0 salience function, agent-based predominant f0 tracking in both frequency bands	joint	2	10 hand labelled excerpts	probabilistic
	Smaragdis and Brown	STFT	NMF	joint	n/a	piano music excerpts	spectrogram-factorisation
	Smaragdis	STFT	NMFD	joint	n/a	drum music excerpts	data-adaptive
	Klapuri	STFT	computation of whitened wide-band spectrum, harmonic selection, subband-weighting	iterative	1–6	mixtures of recorded instrument samples	perceptually-motivated

	Abdallah and Plumbley	STFT	ML estimation of dictionary MAP estimation of activations based on sparsity priors	joint	$\sim$ 2–3	synth. harpsichord + solo piano music	sparsity-based
	Blumensath and Davies	time-domain	OMP-based subspace selection, estimation of activations with log-l_1-constraint, dictionary update	joint	n/a	piano sonata recording	sparsity-based
	Marolt	auditory filterbank	auditory filtering, onset detection by MLPs, partial tracking by adapt. oscillators, TDNNs for note recognition, ad hoc length and loudness estimation	joint	n/a	database of synthesised piano sounds	classification-based
2005	Ryynänen and Klapuri	STFT	multiple-f0 estimation, probabilistic models for notes, silence and note transitions, determination of optimal path through the models	iterative	n/a	RWC music genre database	perceptually-motivated, probabilistic
	FitzGerald et al.	CQT	shift-invariant NMF	joint	n/a	single synthesised duet	data-adaptive
2006	Klapuri	STFT	spectral whitening, f0 salience estimation by weighted summation of partials, direct estimation/iterative cancellation/joint estimation of f0s	direct, joint & iterative	1,2,4 & 6	random and musical mixtures	heuristic
	Schmidt and Mørup	CQT	sparse non-negative matrix factor 2-D deconvolution	joint	2	synth. and recorded piano music	data-adaptive
	Cont	STFT / instant. freq.	template learning from database, real-time decomposition of input spectra in NMF framework with sparsity constraints	joint	n/a	piano and harpsichord pieces from RWC database	data-adaptive
	Davy et al.	Gabor atoms	MAP estimation of the probabilistic model parameters by MCMC sampling	joint	1–4	random mixtures of instrument notes	probabilistic
	Zhou	RTFI	spectral peak-picking, note classification by SVMs with RBF kernels	joint	2–6	random mixtures of monophonic samples	classification-based
2007	Kameoka et al.	CQT	probabilistic model using note models and harmonic temporal clustering	joint	n/a	8 manually annotated single-instrument pieces from RWC database	probabilistic
	Poliner and Ellis	STFT	note classification by SVMs with linear and RBF kernels, HMM postprocessing	joint	n/a	synth. MIDI files and Disklavier piano recordings	classification-based
	Bertin et al.	STFT	NMF analysis, pitch detection of basis functions, activity thresholding	joint	n/a	synth. and recorded piano music	spectrogram-factorisation

	Cont; Cont et al.	STFT	Pre-learning of basis functions, real-time multipitch detection by learning NMF activations with sparsity constraints	joint	n/a	solo piano music from RWC database, artificial violin + piano mixtures	spectrogram-factorisation
	Raczyński et al.	CQT	NMF with harmonicity, sparsity and activity correlation constraints, heuristic onset detection	joint	n/a	four pieces of piano music	spectrogram-factorisation
2008	Pertusa and Iñesta	STFT	noise suppression, pitch salience estimation, salience of f0 combinations, rule-based note tracking	joint	1,2,4 & 6	random mixtures	heuristic
	Vincent et al.	filter-bank analysis on ERB scale	NMF framework with explicit models for basis functions considering different harmonicity and tuning models, spectra are modelled as weighted sum of narrowband spectra	joint	1–7	solo piano music	spectrogram-factorisation
	Niedermayer	STFT	tone model learning, non-negative matrix-division, post-processing of the gain matrix	joint	n/a	single piece of piano music	spectrogram-factorisation
	Saito et al.	Wavelet transform	specmurt analysis, thresholding of note activations	joint	n/a	14 tracks from RWC music database	spectrogram-factorisation
	Cañadas Quesada et al.	harmonic Gabor atoms	HMP, note-event detection method based on spectral smoothness	joint	2–6	random mixtures of instrument sounds, solo piano music	sparsity-based
	Leveau et al.	chirped Gabor atoms	sparse coding with instrument-specific harmonic atoms	iterative	2	2-instrument mixture (flute and clarinet)	sparsity-based
2009	Costantini et al.	CQT	onset detection, note-aligned CQT analysis, note-classification by SVMs with RBF kernels, offset detection procedure	joint	n/a	synth. MIDI files of solo piano music	classification-based
	Mysore and Smaragdis	CQT	shift-invariant PLCA with Dirichlet priors and temporal continuity constraints	joint	2	MIREX development set	spectrogram-factorisation
2010	Cañadas Quesada et al.	STFT	noise thresholding and peak-picking, f0 candidate estimation, exhaustive evaluation of f0-candidate combinations, 2-state HMM post-processing	joint	up to 5	12 pieces from RWC database	heuristic
	Yeh et al.	STFT	noise level estimation, f0 candidate selection, joint f0-estimation, polyphony inference	joint	up to 6	Yeh dataset, MIREX 2007 & 2008	heuristic
	Emiya et al.	STFT	preprocessing to find most salient pitch candidates, ML estimation of parameters for AR spectral envelope model and MA noise model	joint	1–6	MAPS database	probabilistic

	Hennequin et al.	STFT	NMF with parametric basis functions	joint	1–3	single synth. Bach prelude	spectrogram-factorisation
2011	Argenti et al.	CQT	pitch detection by bispectral analysis and pattern matching, independent note duration estimation	iterative	n/a	RWC Classical Music Database	heuristic
	Dressler	multires. STFT	magnitude weighting, peak-picking, pairwise evaluation of peaks based on various criteria, heuristic note tracking	joint	n/a	MIREX 2011 & 2012	heuristic
	Nam et al.	n/a	feature learning by Deep Belief Network, note-classification by SVM with linear kernel, HMM postprocessing	joint	n/a	various datasets of synth. and recorded piano music	classification-based
	Grindlay and Ellis	STFT	NMF with hierarchical eigeninstruments, thresholding for note detection	joint	2–5	MIREX development dataset	spectrogram-factorisation
	Lee et al.	STFT	heuristic note candidate selection, BP with dictionary of note candidate atoms, temporal smoothing	iterative	n/a	solo piano recordings	sparsity-based
2012	Böck and Schedl	STFT	semitone spectrogram and first difference, bidirectional recursive neural network for note classification	joint	n/a	various datasets of synth. and recorded piano music	classification-based
	Grosche et al.	multires. STFT	noise suppression, tuning estimation, salience estimation from semitone spectrogram, separate onset detection, note tracking by HMMs	joint	n/a	synth. audio and real recordings of classical and pop music	heuristic
	Yoshii and Goto	Wavelet transform	infinite latent harmonic allocation (iLHA) and variational Bayes (VB) inference	joint	n/a	Piano and guitar recordings from RWC and MAPS database	probabilistic
	O'Hanlon et al.	STFT	Group sparse representation based on pre-learned piano atoms	joint	n/a	solo piano recordings	sparsity-based
2013	Raczynski et al.	STFT	Pitch salience detection by NMF, dynamic Bayesian network model for chords, note combinations and note saliences	joint	n/a	RWC database	probabilistic
	Benetos and Dixon	CQT	temporally-constrained, shift-invariant PLCA model	joint	n/a	RWC dataset, MIREX developm. set, Disklavier dataset	spectrogram-factorisation

2.2.8 Discussion

The previous sections gave an overview over a variety of approaches for polyphonic music transcription and grouped them into categories that reflect the main underlying analysis methods. Each of the categories exhibit different strengths and shortcomings which are discussed in this section.

Perceptually-motivated approaches

Since human experts can be very adept at transcribing a piece of music, it seems advantageous to model the individual processes of human music perception. However, although the physiological processes of the human auditory system are fairly well researched, perceptually-motivated methods suffer from the problem that the processing stages of the human brain are not yet well understood. Pitch recognition is not an inherent human ability and has to be acquired in early years. Expert musicians who are capable of delivering high-quality transcriptions have aquired this ability through several years of musical training. Hence, not only the actual perceptual processes need to be understood, but also the learning process and the way this information is represented in the human brain.

ASA approaches on the other hand, which, instead of re-building the mechnical processes of the auditory system, model perception by applying findings of psychoacoustic research, are lacking a robust signal representation. Bregman's sinusoidal grouping principles require a representation of sinusoidal tracks. Sinusoidal modelling techniques, however, are rather unreliable and especially fail for more complex musical material with multiple overlapping sinusoidal partials.

Heuristic methods

Heuristic approaches are usually based on certain assumptions about musical instruments, such as harmonicity and spectral smoothness, and about the acoustic environment in which they were recorded. These assumptions are often kept very general in order to enable the detection of notes from a wide variety of musical instruments. This generality is the main strength of heuristic approaches and as a result these methods achieve comparably high transcription accuracies. In the MIREX evaluations from 2007 to 2012, the systems by Yeh et al. (2010) and Dressler (2011) were among the highest ranked algorithms in the *multiple-f0 estimation* and *note tracking* tasks.

On the other hand, heuristic approaches usually detect fundamental frequencies within a recording without any knowledge about the underlying instruments. In fact, techniques such as spectral whitening even deliberately suppress timbral information in order to provide similar transcription results over a wide range of instrument timbres. Due to the fact that harmonically related sounds often

have a large number of overlapping partials (Klapuri, 1998), sounds can become largely obscured. When two simultaneous notes are e. g. an integer number of octaves apart, the partials of the upper note are completely overlapped by the partials of the lower note. This situation is illustrated in Fig. 2.5, where the

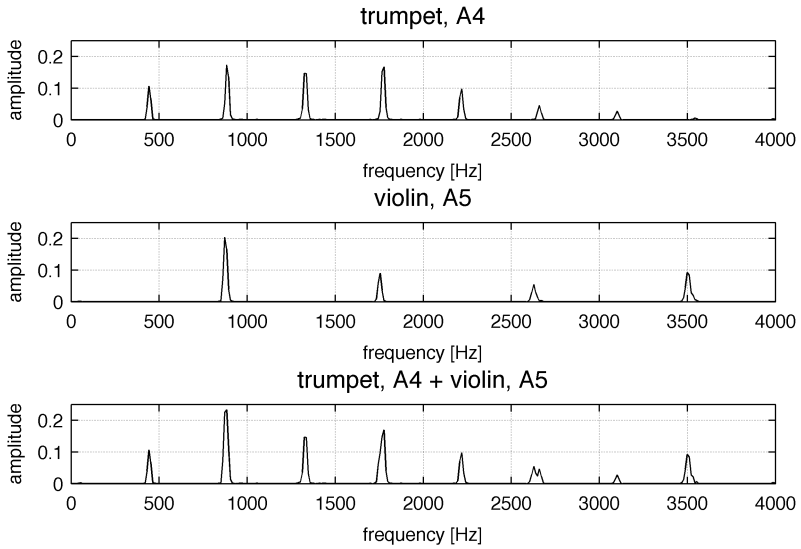


Figure 2.5: Example of overlapping partials. The upper and middle plots display magnitude spectra of a trumpet and a violin tone an octave apart. The lower plot shows the resulting spectrum if both instruments sound simultaneously. The partials of the note with the higher pitch are completely overlapped by those of the note with the lower pitch.

upper and middle diagram display the magnitude spectra of a trumpet sound and a violin sound that are an octave apart. The lower plot shows the spectrum when both instruments sound simultaneously. In this example, any approach that does not take into account timbral information will necessarily fail to detect the upper note. Principles such as the smoothness of the spectral envelope try to address the resolution of overlapping partials, however, this very coarse assumption is likely to fail when the number of overlapping partials is high. Knowledge about typical amplitudes of harmonic partials of the instruments contained in the mixture enables a more accurate decomposition of overlapping partials and the assignment of detected notes to the underlying instruments.

Probabilistic approaches

Probabilistic models are powerful tools for modelling acoustic processes. The strength of these models is their great flexibility and their ability to model

real-world processes in any level of detail. In addition, Bayesian learning theory provides a solid mathematical basis for the inference of the model parameters.

The main difficulty of these methods lies in the design of the model structure and the choice of the Bayesian prior probabilities which need to match the real world data. Particularly the choice of a prior can be difficult when the underlying distribution of the data is not known and sometimes priors are selected for mathematical convenience rather than to match the data.

The computational cost of Bayesian inference can be high depending on the complexity of the model and whether probability distributions are employed in analytical form or as sampled distributions.

Spectrogram-factorisation methods

In recent years, data-adaptive methods have become quite popular for music transcription and source separation tasks. Particularly the introduction of non-negative matrix factorisation made a significant advance, since it provides ways to overcome several of the common problems in music transciption: it estimates the individual spectra of different instruments from the data and thus enables the resolution of overlapping partials. NMF also inherently performs joint estimation of the fundamental frequencies and makes only very few assumptions about the sound sources.

The advantages and shortcomings of the basic NMF algorithm and a few of its extensions are discussed in greater detail in Section 2.4. One of the problems that remain within the NMF framework is the need for a certain amount of prior knowledge about the instrument mixture under analysis. This prior kowledge is necessary to restrict the model in such a way that it extracts musically meaningful information. Spectrogram factorisation methods can also become computationally expensive depending on the size of the spectrogram and the number of basis functions.

Methods based on sparse coding

The area of compressed sensing is based on the notion that the human brain performs redundancy reduction when presented with a complex visual or acoustic scene as indicated by Olshausen and Field (1997). This notion led the authors to formally introduce the concept of sparsity which tries to explain observed data based on a small number of pre-defined hypotheses. For acoustic scene analyses this translates into dictionaries of pre-defined sound spectra and the explanation of a mixture spectrum by only a small number of these spectra.

An inherent problem of sparse methods is the fact that the actual measure for the number of constituent sound spectra is given by the l_0-norm and that

cost functions including the l_0-norm are non-convex and therefore cannot be optimised by gradient methods. For BP methods, the l_1-norm is selected as a replacement for the l_0-norm and the sparsity measure can be weighted against the reconstruction error of the representation. The balance between these two parts is another difficulty in sparse representations. It is not obvious how to select the weight parameter for the sparsity measure, as it depends on several factors, such as the size and structure of the dictionary and the actual sparsity of the analysis spectrogram. It is conceivable that the music to be analysed contains sections with large numbers of simultaneously played notes, which means that the data might not necessarily be sparse at all.

Classification-based approaches

In general, classifiers from the field of machine learning require a training stage in which the model parameters are learnt based on some labelled target data. The choice of the target material plays an important role as it has a major influence on the subsequent performance of the classification algorithm when presented with unseen data samples. If a classifier is used for each individual note of an instrument, the training material needs to contain not only examples of isolated notes, but also examples in which the note occurs in conjunction with other, simultaneous notes of various instruments and pitches. It is not clear what type and what amount of training data is actually required for the classifier to achieve good recognition results in a wide range of musical contexts. It is also not immediately obvious to what extent classifiers are able to recognise spectra from the same type of instrument when recording conditions change.

Pertusa and Iñesta experimented with the cross-detection of different timbres, that is, training the classifier on one timbre and testing it on another. The authors report considerable decreases in performance for some timbre combinations. In a similar way, the system by Böck and Schedl (2012) achieved very low transcription accuracies in the 2012 MIREX evaluation when applied to material other than piano music for which it was trained, which underlines the above mentioned issues.

Connected with the issue of training material selection is the choice of the model complexity for the classifiers. Generally a low order classification model has limited modelling capabilities and might therefore not capture the wide range of possible analysis spectra. If the model complexity is too high on the other hand, the classifier might overfit the training data, which leads to poor generalisation. Methods for estimating an appropriate model order exist, but there is no intuitive way to find out what model order is actually required for the wide range of musical instruments and transcription scenarios.

MIREX evaluation of transcription systems

The *International Music Information Retrieval Systems Evaluation Laboratory*[1] *(IMIRSEL)* at the University of Urbana-Champaign runs annual evaluations of MIR algorithms which enable an objective comparison of the different approaches. The evaluation is called *Music Information Retrieval Evaluation eXchange*[2] *(MIREX)* and covers a wide range of audio content analysis tasks.

The *Multiple Fundamental Frequency Estimation & Tracking* task evaluates music transcription algorithms and has been running every year since 2007. This task is split into three subtasks, namely

1. multiple fundamental frequency estimation (MF0E),
2. note tracking (NT), and
3. instrument tracking.

The second subtask is further divided based on the target audio material into a *mixed set* category consisting of multi-instrument recordings, and a *piano-only* category consisting of solo piano recordings. The third subtask was only run in 2009 and 2010 with just a single participating algorithm. The datasets on which the algorithms were tested slightly varied over the years and included a woodwind recording, quartet recordings of Bach chorales, piano solo performances recorded on a Yamaha Disklavier[3], as well as synthesised MIDI pieces. Polyphonies ranged from 2–5 concurrent pitches. Different evaluation metrics were proposed for the different tasks which included precision and recall measures on a frame and note level.

Many of the highest ranked algorithms over the years belong to the category of heuristic approaches. The systems by Pertusa and Inesta in 2007/2008, Yeh and Röbel from 2007–2011, Zhou and Reiss in 2007/2008 and Dressler in 2011/2012 belong into this category and achieved better results that most other systems for the MF0E subtask. These approaches also did well for NT subtask, but since the results for this task are generally lower, the differences to algorithms from other categories were smaller. Dressler's system also stood out due to its very low computational complexity.

The system by Ryynänen and Klapuri in 2007/2008, which was categorised above in the category of probabilistic approaches also showed very good results in both the MF0E and NT tasks. It employs an acoustic model and a model of musical context. The method by Poliner and Ellis in 2007 did well in the note tracking subtask which was achieved by a classification-based pitch activation

[1] `http://www.music-ir.org/`
[2] `http://www.music-ir.org/mirex/`
[3] `http://usa.yamaha.com/products/musical-instruments/keyboards/disklaviers/`

function and a subsequent note tracking method based on HMMs. Vincent, Bertin and Badeau's approach also achieved good results in both tasks in 2007/2008. It used a harmonic NMF to compute a pitch activation function which was simply thresholded to obtain note activations. In 2009, the transcription algorithm by Nakano, Egashira, Ono and Sagayama outperformed most other submissions. This method was based on the HTC method by Kameoka et al. (2007).

The MIREX results from 2007 to 2013 show that approaches from different categories can achieve competitive results. Many of the highest ranked algorithms are based on heuristic rules and employ general knowledge about harmonic sound sources and the mixture process. As indicated above, however, other approaches might be necessary when instrument models need to be employed in order to track the individual instruments in a recording.

2.3 User-assisted music transcription

Some musicians are highly skilled at transcribing music performances. Hainsworth (2004) conducted an informal study in which he collected the answers of 19 musicians to a questionnaire about their transcription practice. He asked the musicians to identify the individual steps in which they derive a transcription and the order in which those steps are performed. Furthermore, the musicians where asked how accurately they think their final result represents what was actually played in the recording, and how much information they felt they added to it in order to fill gaps. In terms of the transcription accuracy, Hainsworth reports that the answers of the participants could be divided into two different goups: one group who tried to achieve a detailed transcription of what was actually played, and another group whose aim was to maintain the overall characteristics of the piece but who usually added their own arrangements to it — particularly when the identification of compositional details is obscured.

Even though no quantitative studies can be found that indicate the accuracy with which humans are able to transcribe performed pieces of music, there is some intuition about what can be achieved by expert listeners. The transcription of 4-voice chorales, for example, is a common task in many music education programmes and can often be solved with good accuracy by trained human listeners. The transcription of four concurrent sounds, however, is still a challenging task for machine listening algorithms which is confirmed by the annual MIREX evaluation. The advantage of computers on the other hand is their ability to perform tasks quickly and repeatedly. This provokes the question how human perception, knowledge and reasoning can be combined with computational approaches to music transcription in order to achieve results in a shorter amount of time and

with better accuracy. These approaches will be referred to as *user-assisted music transcription*.

In the following section, we will review previous work that employs user-assistance for various audio content analysis tasks. This mainly includes tasks other than music transcription. In Section 2.3.2, we will define criteria for user requests and identify a number of potentially useful types of user information.

2.3.1 Prior work on user assistance

Many proposed transcription systems — often silently — make assumptions about certain parameters, such as the number or types of instruments in the recording (e. g. Dessein et al., 2010; Grindlay and Ellis, 2011; Lee et al., 2011), however, not many systems explicitly incorporate prior information from a human user. Previous work on user-guided music transcription was done by Dittmar and Abeßer (2008) who proposed a system that performs automatic melody, bass, chord and drum transcription. The user is provided with various manipulation options to modify the initial results. The modified results, however, do not feed back into the algorithm and are only used for subsequent editing of the results.

User-assisted techniques have predominantly been applied to the related field of audio source separation. Smaragdis and Mysore (2009) used a hummed melody from the user to separate the corresponding pitches from a mixture. The hummed melody was used as a Dirichlet prior in a PLCA framework. Barry et al. (2004) separated instrument tracks from a stereo recording based on a user-specified azimuth range in the stereo panorama. Ozerov et al. (2012) required information about the number of components per source and a source activity segmentation from the user. The information is employed by setting individual values of the activations in a non-negative tensor factorisation framework to zero. Fuentes et al. (2012) asked the user to select the notes to be separated from an initial automatic transcription result and employed PLCA for the separation. Likewise, Durrieu and Thiran (2012) enable the user to select and modify pitch contours of the main melody to be separated. An approach by Bryan and Mysore (2013) obtains an initial separation of the sources by employing supervised and semi-supervised PLCA. The user is then given the opportunity to annotate certain time-frequency regions in the spectrogram by drawing on a spectrogram display. The annotations are used to refine the initial estimate by applying regularisation on the posterior matrices of the PLCA analysis.

For the task of beat tracking, Dixon (2001) built an interactive system that enabled a user to correct estimated beat times. The user was given the opportunity to re-apply the beat tracking estimation using the corrected beats.

2.3.2 User information

The choice of suitable user information for a user-assisted music transcription system needs to consider both the target user group and the algorithmic requirements. Different user groups can be expected to have different listening experience and musical backgrounds and hence may or may not be able provide certain kinds of information. On the other hand, not all information is immediately beneficial for a transcription algorithm.

User input should fulfill the following criteria in order to be of practical use:

- **Requests should be intuitive and unambiguous.** It should be very clear for the user, what type of information about the mixture is requested. This means that the terminology of the request should match the terminology of the targeted users. This also implies that any aspects of the internal workings of the transcription algorithm should not be part of the request if it cannot be guaranteed that the user knows these details.

- **Users should be able to extract the information easily and reliably.** Not all users have the same musical background and training. Musicologists, musicians, audio engineers and musically inexperienced users all have different degrees of expertise and will not interpret musical content in the same way. It is important to consider the target user group and make sure that this group is capable of providing the requested information.

- **Providing the information should not take too much time and effort.** For a practical application it is important that a user can provide the information in a limited amount of time and with reasonable effort. This implies that tedious tasks such as large-scale annotations should be avoided.

Given these criteria, the following types of information could potentially be obtained from a user:

- key, tempo and time signature of a piece,
- number and types of instruments in the recording,
- information about repeated structural segments and musical motifs,
- providing a few chord labels,
- providing annotations of a small number of notes for each instrument.

This list is by no means exhaustive.

From the above list it becomes clear that different degrees of musical expertise are required to provide different types of information. While naming the instruments in the recording or providing structural information might even be achieved by less-trained listeners, providing chord labels requires more thorough musical training and experience. Also, users might have to revert to additional means in order to be able to provide the information: an objective tempo estimate can only be given when a metronome is at hand, and musicians without absolute pitch will need a reference pitch in order to correctly identify key or chords.

From an algorithmic perspective, not all user information is equally useful for the transcription task and a decision must be made about what kind of information can be employed to enhance the performance of the transcription algorithm. Knowledge about repeated sections could for example be used to facilitate the transcription of segments based on information obtained at a corresponding part. Key or chord information could be useful in order to eliminate spurious pitch candidates.

The various analysis techniques outlined in the literature review in Section 2.2 have different preconditions to incorporate user information. Probabilistic models for example usually integrate knowledge as Bayesian priors but might also apply conditional modifications to the model structure. Other methods such as classification-based approaches have less scope for prior knowledge.

Our investigations in this thesis will focus on information that facilitates the creation of accurate *instrument* and *timbre models* for the initial low level analysis of audio recordings. While automatic timbre identification in polyphonic music is a challenging task, human listeners are often capable of identifying the instruments in a recording or associating certain notes or parts with a specific instrument. Knowledge about instrument timbre is necessary if a transcription of the individual instrument parts from a recording is required. Fully automatic transcription systems typically only focus on the extraction of note events without considering the instrument assignments of the notes.

Timbre information can well be incorporated in a non-negative matrix factorisation model. These models enable the representation of musical instruments by a set of prototypical sound spectra at various pitches. Timbre is here encoded in the amplitudes of the harmonic partials of the spectra which differ for the spectra of different instruments at the same pitch. An introduction to non-negative matrix factorisation and some of its variants will be provided in the following section.

2.4 Non-negative matrix factorisation techniques

The analysis framework for user-assisted music transcription in this thesis is based on the non-negative matrix factorisation (NMF) technique. Non-negative matrix factorisation has become quite popular for audio analysis purposes within the last decade. The basic principles were already described by Paatero and Tapper (1994) in 1994 as *positive matrix factorisation*, but NMF gained considerable attention through an article by Lee and Seung in the *Nature* journal in 1999. Since then NMF has been adapted to various engineering domains that deal with non-negative data, such as audio processing, image processing and information retrieval.

The strength of the NMF algorithm for audio analysis lies in its ability to capture characteristics of the instruments from the recording itself without incorporating explicit parametric instrument models. In addition, NMF makes only very few assumptions about the underlying sources, which provides the opportunity to adapt to a variety of instrument types. Certain parameters however still have to be estimated and set in advance and we discuss the difficulties that arise from these estimates in the following section. For the task of semi-automatic music transcription, NMF is a very useful tool because it provides simple and effective ways to accommodate prior information about the sound sources and about the occurrence of musical events by setting the initial conditions for the NMF analysis.

This section will introduce NMF, provide some intuition about its analysis results, explain its relation to other techniques and introduce some of its extensions.

2.4.1 Standard non-negative matrix factorisation

The standard non-negative matrix factorisation technique was introduced by Lee and Seung in 1999 and was first applied to music analysis by Smaragdis and Brown in 2003. NMF is — just as Principal Component Analysis (PCA) and Independent Component Analysis (ICA) — a matrix factorisation technique. It decomposes a matrix into the product of two other matrices that both have a lower rank than the original matrix. While PCA and ICA estimate the basis functions — that is the columns of the first, lower-rank matrix — by enforcing maximum variance and statistical independence, respectively, NMF imposes a non-negativity constraint on the matrices.

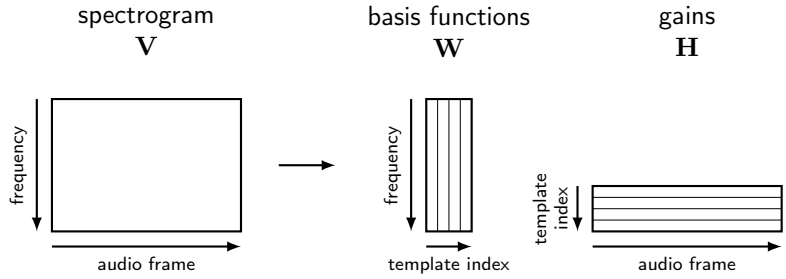


Figure 2.6: Schematic illustration of NMF.

Model

The matrix factorisation is expressed by

$$\mathbf{V} \approx \mathbf{\Lambda} = \mathbf{W} \cdot \mathbf{H}, \tag{2.2}$$

where $\mathbf{V}$ denotes the original matrix, $\mathbf{W}$ and $\mathbf{H}$ the low-rank approximations, and $\cdot$ indicates matrix multiplication. $\mathbf{\Lambda}$ is used to account for the fact that the matrix product will only yield an *approximation* of the original matrix due to the information loss introduced by the rank reduction. A visualisation of the matrix factorisation and the dimensions of each matrix is shown in Fig. 2.6.

Two ambiguities apply to all types of matrix factorisations:

Scale ambiguity A multiplication of any of the column vectors in $\mathbf{W}$ with a factor c_r and a multiplication of the corresponding row of $\mathbf{H}$ by the inverse of the factor results in the same matrix $\mathbf{\Lambda}$. If we express $\mathbf{W}$ and $\mathbf{H}$ as concatenations of vectors as follows

$$\mathbf{W} = \begin{bmatrix} | & | & & | \\ \mathbf{w}_0 & \mathbf{w}_1 & \dots & \mathbf{w}_{R-1} \\ | & | & & | \end{bmatrix}, \qquad \mathbf{H} = \begin{bmatrix} — & \mathbf{h}_0^\top & — \\ — & \mathbf{h}_1^\top & — \\ & \vdots & \\ — & \mathbf{h}_{R-1}^\top & — \end{bmatrix}, \tag{2.3}$$

the ambiguity can be expressed as

$$\mathbf{\Lambda} = \mathbf{W} \cdot \mathbf{H} = \sum_{r=0}^{R-1} \mathbf{w}_r \cdot \mathbf{h}_r^\top = \sum_{r=0}^{R-1} c_r \mathbf{w}_r \cdot \frac{1}{c_r} \mathbf{h}_r^\top. \tag{2.4}$$

In these equations, the operator $\top$ denotes vector transposition.

Permutation ambiguity Any permutation of the columns in $\mathbf{W}$ and the same permutation of the rows in $\mathbf{H}$ leave the resulting matrix $\mathbf{\Lambda}$ unaltered.

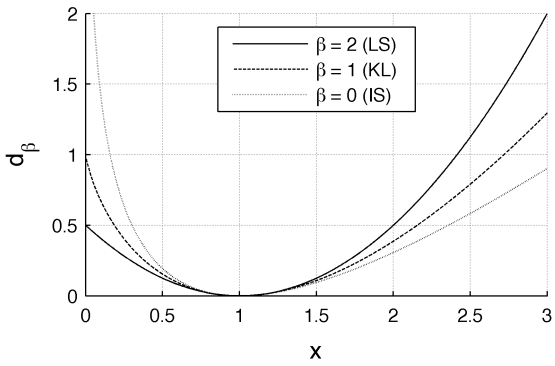


Figure 2.7: β-divergence $d_\beta(x, y)$ for different values of β, and $y = 1$.

Cost functions

In order to approximate the matrices $\mathbf{W}$ and $\mathbf{H}$, a cost function C between the original matrix $\mathbf{V}$ and its approximation $\mathbf{\Lambda}$ is minimised. Commonly used cost functions are based on the *least squares error (LS)*, the *Kullback-Leibler divergence (KL)* and the *Itakura-Saito divergence (IS)* of the matrix elements. The β-divergence (Cichocki et al., 2006) combines the three in a single error measure. The β-divergence between two real, non-negative values x and y is given by

$$d_\beta(x, y) = \frac{x^\beta}{\beta(\beta - 1)} + \frac{y^\beta}{\beta} - \frac{xy^{\beta-1}}{\beta - 1}, \tag{2.5}$$

for $\beta \in \mathcal{R}\backslash\{0, 1\}$, and

$$d_0(x, y) = d_{IS}(x, y) = \frac{x}{y} - \log\left(\frac{x}{y}\right) - 1 \tag{2.6}$$

$$d_1(x, y) = d_{KL}(x, y) = x \cdot \log\left(\frac{x}{y}\right) + y - x. \tag{2.7}$$

The least squares cost function is given for $\beta = 2$, the Kullback-Leibler divergence by $\beta = 1$, and the Itakura-Saito divergence by $\beta = 0$. Figure 2.7 illustrates these divergences.

An interesting property of the IS divergence is its invariance towards scalings of the arguments (Févotte et al., 2009):

$$d_{IS}(\gamma \cdot x, \gamma \cdot y) = d_{IS}(x, y) \tag{2.8}$$

The IS divergence thus measures the *relative* divergence between the values x and y, which means that in order to produce the same divergence, the difference between x and y needs to be larger for larger absolute values of x and y than for smaller absolute values. This is in compliance with the Weber-Fechner law of

perception (Fechner, 1860) which states that a perceived difference in loudness is proportional to the absolute value of the stimulus. The scale-invariance property, however, also makes the IS divergence sensitive to deviations of very small amplitudes (e. g. the noise floor). Theoretical work on NMF (Virtanen et al., 2008; Févotte et al., 2009) associate the KL-divergence with magnitude spectra and the IS divergence with power spectra for statistical coherence. The Weber-Fechner law, however, also justifies the use the IS divergence in combination with magnitude spectra.

The NMF cost function sums all divergences of the individual matrix elements:

$$C_\beta = \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} d_\beta(\mathbf{V}_{k,n}, \mathbf{\Lambda}_{k,n}). \tag{2.9}$$

In this equation, k and n denote the row and column index of the matrices, and K and N the number of rows and columns, respectively.

Update equations

The minimisation of the cost functions is achieved by applying gradient descent to the matrices $\mathbf{W}$ and $\mathbf{H}$:

$$\mathbf{W} \leftarrow \mathbf{W} - \eta_W \frac{\partial C_\beta}{\partial \mathbf{W}}, \tag{2.10}$$

$$\mathbf{H} \leftarrow \mathbf{H} - \eta_H \frac{\partial C_\beta}{\partial \mathbf{H}}, \tag{2.11}$$

where $\frac{\partial C_\beta}{\partial \mathbf{W}}$ and $\frac{\partial C_\beta}{\partial \mathbf{H}}$ denote the gradient of C_β w.r.t. $\mathbf{W}$ and $\mathbf{H}$, respectively, and η_W and η_H the step sizes. The convergence of gradient descent methods towards a local minimum of the cost function is strongly dependent on the step size and the topology of the cost function. A major contribution of Lee and Seung's original NMF publication (2001) is the choice of a non-uniform step size that transforms the additive update rules in Eqs. 2.10 and 2.11 into multiplicative update rules. For the general case of the β-divergence, these are given by (Févotte et al., 2009):

$$\mathbf{W} \leftarrow \mathbf{W} \bullet \frac{\left(\mathbf{V} \bullet \mathbf{\Lambda}^{\beta-2}\right) \mathbf{H}^\top}{\mathbf{\Lambda}^{\beta-1} \mathbf{H}^\top} \tag{2.12}$$

$$\mathbf{H} \leftarrow \mathbf{H} \bullet \frac{\mathbf{W}^\top \left(\mathbf{V} \bullet \mathbf{\Lambda}^{\beta-2}\right)}{\mathbf{W}^\top \mathbf{\Lambda}^{\beta-1}}. \tag{2.13}$$

In these equations, $\bullet$ denotes elementwise multiplications and the divisions as well as the exponentiations are likewise performed elementwise. The operator $\top$ denotes matrix transposition. Due to the non-negativity of all matrices on the right-hand side of Eqs. 2.12 and 2.13, the matrix elements of $\mathbf{W}$ and $\mathbf{H}$ are

multiplied by non-negative factors and thus remain non-negative. The derivation of the multiplicative update rules proposed by Lee and Seung (2001) is based on an auxiliary function that guarantees that the cost is reduced in each iteration and that the cost function converges towards a local minimum.

Application to audio analysis

When NMF is applied to audio analysis, the matrix $\mathbf{V}$ corresponds to the magnitude or power spectrogram, i. e. the magnitudes or powers of the STFT or of a constant-Q analysis with uniformly spaced frames. The matrix $\mathbf{W}$ should ideally contain in its columns the most common *spectral shapes* (*basis functions* or *templates*) of the spectrogram and $\mathbf{H}$ should contain in its rows the *gains* (*activations*) of each spectral shape. The gains represent the scaling factors of the basis functions at each time frame. Both matrices are usually initialised with non-negative random values and the algorithm runs either for a fixed number of iterations or until a certain termination criterion is met.

The standard NMF algorithm assumes that the basis functions add linearly, which is not strictly true for magnitude spectrograms. Although the linearity assumption holds for both the STFT and the constant-Q transform in the complex domain, it only holds for the magnitudes if the summed components are in phase, which is usually not the case. However, in many cases it seems to be a reasonable approximation.

When applying the NMF algorithm, one difficulty is that the number of basis functions, i. e. the rank of the matrices $\mathbf{W}$ and $\mathbf{H}$, needs to be set in advance. Problems arise when the rank is not chosen accurately: too many components result in basis functions that might explain only parts of the note spectra, e. g. a subset of partials of a harmonic spectrum. If the number of basis functions is too small on the other hand, the algorithm will not have enough expressivity to achieve an accurate decomposition and will either omit important components or combine separate entities into the same basis function. Typically, the number of distinct pitches of each instrument as well as the number of instruments in the target audio is not known in advance. Furthermore, components might not only correspond to harmonic spectra of different pitches, but can also represent spectra of transients or continuous noise-like sounds of an instrument which further complicates the estimation of the optimal number of components.

Further difficulties arise for instruments that are able to produce pitches on a continuous scale, such as most wind and string instruments. In order to accurately represent a *glissando* or a note with *vibrato*, a large number of components would be required which conflicts with the low rank approximation assumption of the algorithm.

In any case, in order to derive a transcription, several postprocessing steps need to be applied: for each derived basis function a decision must be made whether it represents a tonal or noise-like component. Components classified as tonal need to be further analysed for their pitch. Additionally, if a parts-based transcription is aimed for, an assignment of each tonal spectrum to an instrument is required. These post-processing steps make it clear that although NMF is capable of extracting underlying instrument spectra with little knowledge about the instruments, instrument models of some form are still required to interpret and group the extracted basis functions.

2.4.2 Visualisations

In this section, we will provide some geometrical interpretations of different aspects of the NMF method. These are meant to provide some more intuition and insights into this procedure. This overview was inspired by the work of Shashanka (2008) on geometric interpretations of latent variable models.

Basis function projections

The NMF basis functions can be interpreted as vectors in a K-dimensional space. Since all vector components are non-negative, all basis functions live in the first hyperoctant of this K-dimensional space. The scale ambiguity described in Section 2.4.1 is usually eliminated by normalising the basis functions according to the l_1 (Manhattan) norm or by the l_2 (Euclidean) norm and multiplying the gains accordingly. This normalisation corresponds to a projection of all basis vectors onto a hyperplane or the surface of a hypersphere in the first hyperoctant. Figure 2.8 displays the hyperplane (also denoted as *simplex*) and the hypersphere onto which all basis functions are projected by the normalisation for a dimensionality of $K = 3$.

Subspaces

The basis functions are expected to capture the typical spectral shapes of the underlying spectra of the music mixture. Since the basis functions can be individually scaled in each time frame by their corresponding gains, only the *directions* of the basis function vectors in the K-dimensional space are of interest. The *simplex* representation (see Fig. 2.8) introduces an intuitive way to visualise the directions of spectra without considering the lengths of both the basis function vectors and the vectors of the underlying instrument spectra Shashanka (2008).

The entirety of vectors that can be represented by weighted combinations of the basis functions is denoted as a *subspace*. We can visualise the subspace

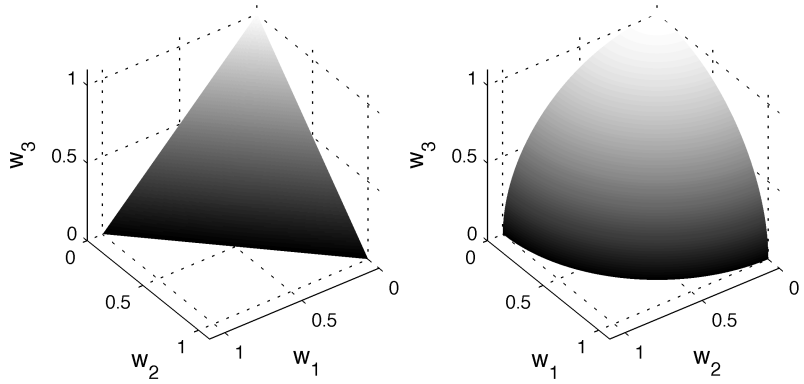


Figure 2.8: Projection surfaces for normalised basis functions for a dimensionality of $K = 3$. Left: simplex (l_1-norm), right: hypersphere (l_2-norm).

in the simplex representation by displaying only those directions in which the subspace vectors can point.

Figure 2.9 illustrates the subspaces spanned by different numbers of basis functions on the simplex for a dimensionality of $K = 3$. Two basis functions (Fig. 2.9 left) span a subspace that corresponds to a single line segment. That means that only instrument spectra that lie on this line segment can be accurately represented by these basis functions. All other instrument spectra will be approximated with a certain degree of error. The subspace spanned by three basis vectors (Fig. 2.9 centre) covers the triangular area between the vectors. For four vectors (Fig. 2.9 right), the subspace corresponds to a tetragon. In general the subspace lies within the *convex hull* of the basis vectors, that is, the polygon that is spanned by the basis vectors.

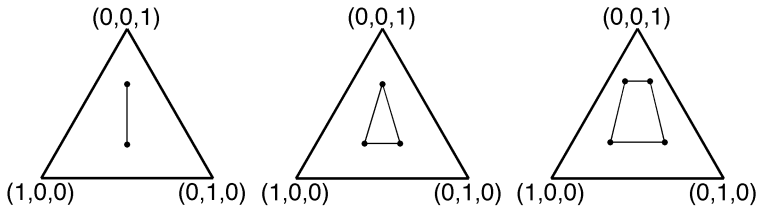


Figure 2.9: Simplex representation of subspaces spanned by different numbers of basis functions. Left: 2 basis functions, centre: 3 basis functions, right: 4 basis functions. The markers represent the basis functions and the lines represent the vertices of the convex hull spanned by the basis functions.

As mentioned in Section 2.4, NMF minimises the cost function between the original spectrogram and its approximation by the weighted basis functions. Since only spectra within the convex hull of the basis functions can be accurately appoximated, NMF estimates the basis function in a such way that the data is *inscribed* in the convex hull of its basis functions. Figure 2.10 illustrates a toy example of some instrument spectra drawn from three i.i.d. multivariate Gaussian distributions with different mean vectors, along with the computed NMF basis functions and their convex hull. It can be seen that the triangle formed by the basis functions includes the majority of data points. In this

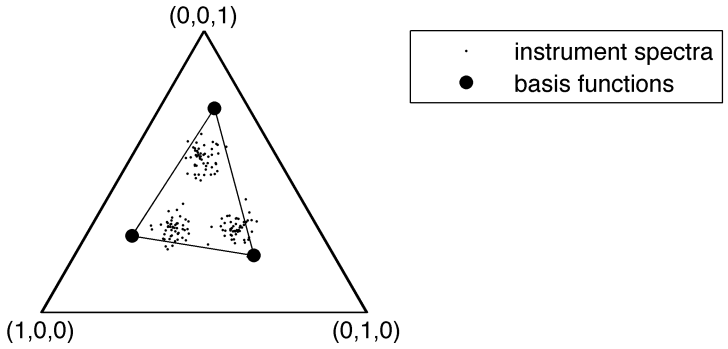


Figure 2.10: NMF basis functions for non-negative data drawn from three multivariate Gaussian distributions.

figure, the basis functions are relatively close to the Gaussian cluster centers and therefore carry some semantic meaning. However, this is not guaranteed and it is highly dependent on the initialisation of the basis functions. Figure 2.11 shows another set of basis functions for the exact same data. Here it becomes clear that even though the data can be well approximated by the basis functions, it is not guaranteed that the basis functions contain meaningful semantic information, as they cannot easily be associated with the means of the three Gaussians.

2.4.3 Variants of NMF

2.4.3.1 Convolutive NMF

In 2004, Smaragdis proposed an extension of the standard NMF algorithm that enables the use of *time-extended basis functions*. Instead of using single frame spectral templates as described in Section 2.4.1, the basis functions now represent *spectrogram fragments*. This method was initially denoted as *non-negative matrix factor deconvolution (NMFD)* but is more commonly known as *convolutive NMF* (O'Grady and Pearlmutter, 2006).

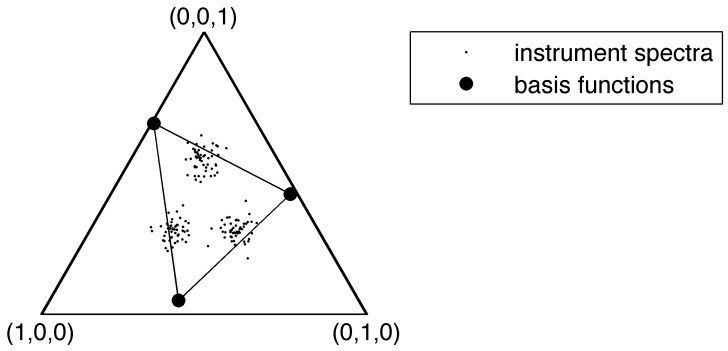


Figure 2.11: Another set of basis functions for the same data as in Fig. 2.10. These basis functions cannot easily be associated with the Gaussian clusters.

Model

Convolutive NMF assumes that all time-extended basis functions are of the same same length $\mathcal{T}$. The matrix $\mathbf{W}$ of the standard NMF method is thus replaced by a number of matrices $\mathbf{W}^\tau$ with $\tau \in [0, \ldots, \mathcal{T}-1]$, one for each time frame of the time-extended basis functions. These matrices can be thought of as a tensor, i. e. a stack of matrices. The mixture model is given by

$$\mathbf{V} \approx \mathbf{\Lambda} = \sum_{\tau=0}^{\mathcal{T}-1} \mathbf{W}^\tau \cdot \overset{\tau\rightarrow}{\mathbf{H}} \tag{2.14}$$

In this equation, the operator $\tau \rightarrow$ denotes a shift of the components of the matrix $\mathbf{H}$ by τ matrix indices to the right while filling the τ leftmost columns with zeros. This shift ensures that the same gain value is applied to all successive frames of the time-extended basis functions. Figure 2.12 illustrates the matrices of Eq. (2.14).

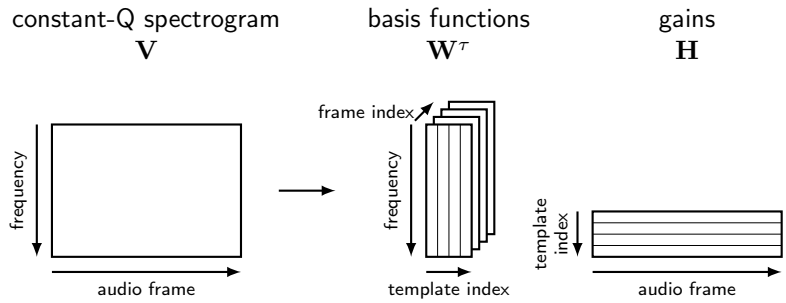


Figure 2.12: Schematic illustration of convolutive NMF.

Update equations

Smaragdis provided multiplicative update equations only for the Kullback-Leibler divergence cost function:

$$\mathbf{W}^{\tau} \leftarrow \mathbf{W}^{\tau} \bullet \frac{\frac{\mathbf{V}}{\mathbf{\Lambda}} \cdot \overset{\tau\rightarrow}{\mathbf{H}}^{\top}}{\mathbf{1} \cdot \overset{\tau\rightarrow}{\mathbf{H}}^{\top}} \tag{2.15}$$

$$\mathbf{H} \leftarrow \mathbf{H} \bullet \frac{\mathbf{W}^{\tau\top} \cdot \overset{\tau\leftarrow}{\left(\frac{\mathbf{V}}{\mathbf{\Lambda}}\right)}}{\mathbf{W}^{\tau\top} \cdot \mathbf{1}} \tag{2.16}$$

In these equations, $\mathbf{1}$ is a matrix of ones with the same dimensions as $\mathbf{V}$ and $\mathbf{\Lambda}$. Smaragdis proposed to iteratively update $\mathbf{H}$ and $\mathbf{W}^{\tau}$ for each value of τ. After the algorithm has terminated, $\mathbf{H}$ should contain a peak at the first time frame at which each basis function occurs.

Application to audio analysis

The time extension of the basis functions enables the retrieval of musical events with a *fixed length* and a recurring time-frequency structure. The scope of convolutive NMF thus lies mainly in the detection of percussive sounds for which these criteria might apply. It might however also be applied to pitched sounds that can be expected to have a fixed length such as harpsichord tones.

The problem of setting the number of components in advance arises for convolutive NMF in the same way as for standard NMF. Additionally, the length $\mathcal{T}$ of the basis functions needs to be estimated. Since all time-extended basis functions have the same length, shorter sounds are represented by too many frames. This might lead to unpredictable values in the remaining time frames after the relevant content in the basis functions.

2.4.3.2 Shift-invariant NMF

Shift-invariant NMF (siNMF) builds on the assumption that the spectra of a musical instrument can be characterised by a fixed spectral shape, i. e. a set of fixed partial amplitudes, usually within a restricted fundamental frequency range. The method was named *shifted NMF* by FitzGerald et al. (2005), but it is better known by the term *shift-invariant NMF* which was introduced by Smaragdis and Raj (2007).

The algorithm operates on a constant-Q spectrogram which has the inherent property that the distances between adjacent partials of perfectly harmonic spectra are not dependent on the fundamental frequency. This implies that a translation of a harmonic spectrum along the frequency axis results in a valid harmonic structure at a different fundamental frequency. Shift-invariant NMF

estimates a number of fixed spectral shapes that are valid within limited pitch ranges.

Model

FitzGerald et al. formulate the algorithm in tensor notation. In order to keep the notation consistent with the methods presented in the previous sections, a notation adapted from Schmidt and Mørup (2006a) is presented here. The mixture is decomposed as follows:

$$\mathbf{V} \approx \mathbf{\Lambda} = \sum_{\phi=0}^{\Phi-1} \overset{\phi\downarrow}{\mathbf{W}} \cdot \mathbf{H}^{\phi} \tag{2.17}$$

Similar to the model in Section 2.4.3.1, $\phi\downarrow$ represents a downward shift of the rows in matrix $\mathbf{W}$ while filling the topmost ϕ rows with zeros. In this model, the matrices $\mathbf{H}^{\phi}$ with $\phi \in [0, \ldots, \Phi - 1]$ contain the gains for the different shifts of the basis functions in $\mathbf{W}$ and can be seen as a tensor. All matrices are illustrated in Fig. 2.13.

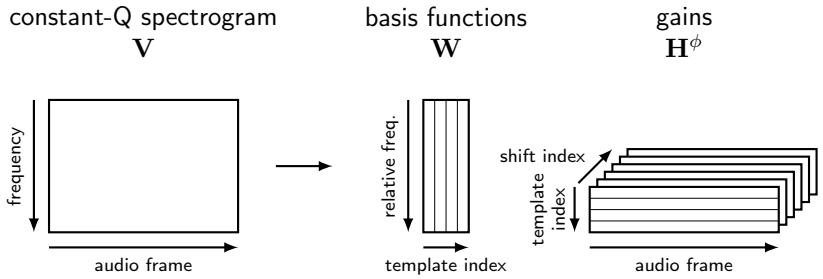


Figure 2.13: Schematic illustration of shift-invariant NMF.

Update equations

The update equations by FitzGerald et al. (2005) are only provided for the KL-divergence:

$$\mathbf{W} \leftarrow \mathbf{W} \bullet \frac{\sum_{\phi=0}^{\Phi-1} \left(\frac{\mathbf{V}}{\mathbf{\Lambda}}\right) \cdot \overset{\phi\uparrow}{\mathbf{H}^{\phi}}^{\top}}{\sum_{\phi=0}^{\Phi-1} \mathbf{1} \cdot \mathbf{H}^{\phi\top}} \tag{2.18}$$

$$\mathbf{H}^{\phi} \leftarrow \mathbf{H}^{\phi} \bullet \frac{\overset{\phi\downarrow}{\mathbf{W}}^{\top} \cdot \left(\frac{\mathbf{V}}{\mathbf{\Lambda}}\right)}{\overset{\phi\downarrow}{\mathbf{W}}^{\top} \cdot \mathbf{1}} \tag{2.19}$$

Application to audio analysis

Shift-invariant NMF is a step towards accommodating pitch variations on a more fine grained pitch scale. Pitch variations are an inherent musical property and need to be addressed if an accurate representation is aimed for. However, partial amplitudes can vary for different pitches of the same instrument. A single spectral shape, shifted across the whole fundamental frequency range is only a very coarse model for a musical instrument.

2.4.3.3 NMF2D

A combination of shift-invariant and convolutive NMF was presented by Schmidt and Mørup under the name *non-negative matrix factor 2-D deconvolution (NMF2D)*. It combines time-extended basis functions with shifts along the frequency axis and thus allows spectrogram fragments to be detected both in the time and frequency dimension.

Model

The model is defined as follows:

$$\mathbf{V} \approx \mathbf{\Lambda} = \sum_{\tau=0}^{\mathcal{T}-1} \sum_{\phi=0}^{\Phi-1} \overset{\phi\downarrow}{\mathbf{W}}^{\tau} \cdot \overset{\tau\rightarrow}{\mathbf{H}}^{\phi} \tag{2.20}$$

It contains both a frequency shift operator $\phi\downarrow$ and a time shift operator $\tau\rightarrow$. The set of matrices $\mathbf{W}^{\tau}$ as well as the set of matrices $\mathbf{H}^{\phi}$ can be seen as tensors. The spectrogram as well as the tensor structures for basis functions and gains are illustrated in Fig. 2.14.

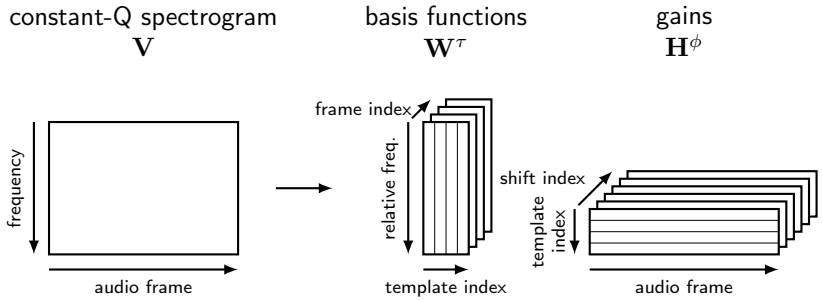


Figure 2.14: Schematic illustration of NMF2D.

Update equations

Schmidt and Mørup provide update equations for the least squares error cost function as well as the Kullback-Leibler divergence. Here, only the latter are presented:

$$\mathbf{W}^{\tau} \leftarrow \mathbf{W}^{\tau} \bullet \frac{\sum_{\phi=0}^{\Phi-1} \left(\frac{\mathbf{V}}{\mathbf{\Lambda}}\right) \cdot \overset{\tau\rightarrow}{\mathbf{H}}^{\top}}{\sum_{\phi=0}^{\Phi-1} \mathbf{1} \cdot \overset{\tau\rightarrow}{\mathbf{H}}^{\top}} \tag{2.21}$$

$$\mathbf{H}^{\phi} \leftarrow \mathbf{H}^{\phi} \bullet \frac{\sum_{\tau=0}^{\mathcal{T}-1} \overset{\phi\downarrow}{\mathbf{W}}^{\top} \cdot \left(\frac{\mathbf{V}}{\mathbf{\Lambda}}\right)^{\tau\leftarrow}}{\sum_{\tau=0}^{\mathcal{T}-1} \overset{\phi\downarrow}{\mathbf{W}}^{\top} \cdot \mathbf{1}} \tag{2.22}$$

Application to audio analysis

Although NMF2D seems to be the most versatile extension of NMF as it combines time-extended basis functions with frequency shifts, its practical relevance is actually lower than convolutive NMF and shift-invariant NMF. Per definition, unpitched instrument sounds — for which the convolutive NMF model is a good fit — do not need to be shifted to different pitches along the frequency axis. Pitched sounds on the other hand are usually not well described by fixed-length spectrogram fragments as note lengths vary throughout a piece of music. Using single frame templates has the advantage that all temporal information is captured in the gain matrix.

2.5 Viterbi decoding

For pitch tracking in Chapter 5 we employ the Viterbi algorithm to find the most likely sequence of pitches based on pitch hypotheses in each time frame. The Viterbi algorithm is a method to find the most likely state sequence of a discrete first order Markov chain and was introduced by Viterbi in 1967. Below, we summarise its general principles based on the description by Rabiner (1989) in his tutorial on HMMs.

A discrete Markov process is based on the assumption that a system can be in one of N_S distinct states S_x. The actual state of the systems at time n is denoted by q_n. In a first order Markov model, the probability of transition to state S_x at time n only depends on the state S_y at the previous time frame $n-1$,

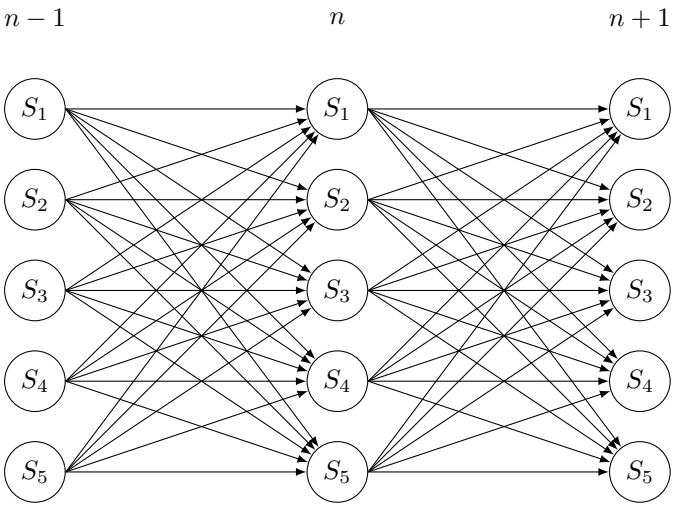


Figure 2.15: All possible state sequences of the Viterbi algorithm for three consecutive time frames.

and this probability is denoted by $p(q_n = S_x|q_{n-1} = S_y)$. Given a sequence of states $O = \{q_1, \ldots, q_N\}$, the overall probability of this sequence is given by

$$p(O) = p(q_1) \cdot p(q_2|q_1) \cdot p(q_3|q_2) \cdot \ldots \cdot p(q_N|q_{N-1}). \tag{2.23}$$

$p(q_1)$ denotes the prior probability of the first observed state q_1. Figure 2.15 displays all possible state sequences for three consecutive time frames for a model with $N_S = 5$ states.

If the state transition probabilities are known, we can ask for the state sequence with the highest probability. Since there are a total number of ${N_S}^N$ combinations, an exhaustive evaluation of all sequence probabilities is not feasible for long sequences and large numbers of states. The Viterbi algorithm provides a way of finding the most likely sequence by making use of the dynamic programming paradigm.

In order to find the most likely state sequence, a term $\delta_n(x)$ is introduced as

$$\delta_n(x) = \max_{q_1, q_2, \ldots, q_{n-1}} \{p(q_1, q_2, \ldots, q_{n-1}, q_n = S_x)\}. \tag{2.24}$$

It describes the probability of the most likely state sequence that ends in state S_x at time n. The most likely state sequence at the successive time frame is then given by

$$\delta_{n+1}(y) = \max_x \{\delta_n(x) \cdot p(q_{n+1} = S_y|q_n = S_x)\}. \tag{2.25}$$

Another term $\psi_n(x)$ keeps track of the states that have been visited up to state S_x at time n.

The Viterbi algorithm then performs the following steps:

- **Initialisation:** Initialise $\delta_1(x)$ and $\psi_1(x)$ for $x \in \{1, \ldots, N_S\}$ by

$$\delta_1(x) = p(S_x) \tag{2.26}$$

$$\psi_1(x) = 0. \tag{2.27}$$

- **Recursion:** For each successive time frame $n \in \{2, \ldots, N\}$, compute $\delta_n(y)$ for $y \in \{1, \ldots, N_S\}$ according to

$$\delta_n(x) = \max_x \{\delta_n(x) \cdot p(q_n = S_y | q_{n-1} = S_x)\}, \tag{2.28}$$

  and keep track of the previous state index

$$\psi_n(y) = \operatorname*{argmax}_x \{\delta_n(x) \cdot p(q_n = S_y | q_{n-1} = S_x)\}. \tag{2.29}$$

- **Termination:** After the computation of all $\delta_N(x)$ at the last time frame for $x \in \{1, \ldots, N_S\}$, the probability of the most likely state sequence is given by

$$p^* = \max_x \{\delta_N(x)\}. \tag{2.30}$$

  The index of the last state is given by

$$q_N^* = \operatorname*{argmax}_x \{\delta_N(x)\}. \tag{2.31}$$

- **Backtracking:** Successively, in reverse order of time, follow the state indices that led to the sequence with the highest probability:

$$q_n^* = \psi(q_{n+1}^*), \tag{2.32}$$

  for $n \in \{N-1, N-2, \ldots, 1\}$.

Even though the Viterbi algorithm was above illustrated with a fixed number of states and constant transition probabilities, it can likewise be applied to systems in which the number of states as well as the state transition probabilities change from frame to frame. An example of this scenario is illustrated in Fig. 2.16.

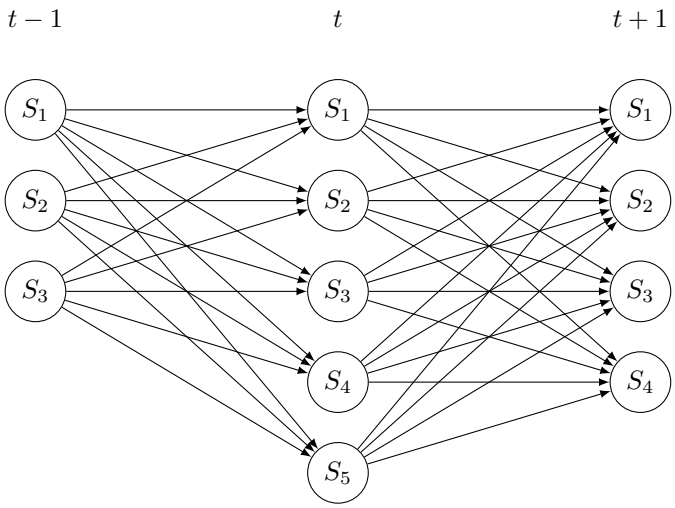


Figure 2.16: All posssible Viterbi state sequences with a variable number of states per frame.

Chapter 3

User-assisted extraction of timbre models

Criteria for potential user information were introduced in Section 2.3.2 and a few examples of information that could be obtained from a musically-trained user were provided. In this chapter, several types of user information are explored that can be used to extract *timbre information* about the instruments in the recording. Timbre models are required for parts-based transcriptions of multi-instrument recordings in which detected notes need to be assigned to the individual instrument sources. The extracted timbre models are employed for an initial low level analysis of the pitch content and their performance is evaluated and compared. Conclusions are drawn about the most beneficial type of user information for the initial transcription process.

In the following section, a brief overview over the research on the timbre of pitched musical instruments will be given. Section 3.2 introduces the non-negative analysis framework which is employed in subsequent chapters. It enables the incorporation of several types of user information. In Section 3.3, two different types of user information are considered that enable the use of different timbre models. The transcription accuracy achieved by both types of models are experimentally evaluated. In Section 3.4, a method for the extraction of a more refined timbre model is proposed that enables modelling a larger variety of instrument sounds.

3.1 Timbre of musical instruments

The timbre of a musical instrument is often denoted as its *sound quality*. It allows human listeners to perceptually distinguish the sounds of different instruments.

The acoustical properties of a sound that determine its perceived timbre have interested researchers for more than a century. A good overview over this research was provided by Risset and Wessel (1999).

Von Helmholtz (1870, 1954) was the first to find that the timbre of an instrument is largely dependent on the amplitude relations of the harmonic partials, which can be observed by a Fourier analysis of the steady state part of the sound. Although Helmholtz also found that "certain characteristic peculiarities in the tones of several instruments depend on the mode in which they begin and end", he deliberately focused on "the peculiarities of the musical tone which continues uniformly". Two decades later, Stumpf (1890) emphasised that the length and character of sound onsets and offsets as well as additional sounds and noises also determine the timbre of an instrument. Analyses in the first half of the 20th century (Meyer and Buchmann, 1931; Hall, 1937), however, focused on average spectra of the instruments, not least due to the available sound analysers at that time. According to Risset and Wessel "it was believed that this difference in average spectrum was utterly responsible for timbre differences" and that "this view is still widely accepted".

Von Helmholtz also stated that the timbre was "solely dependent on the presence and strength of partial tones, and in no respect on the differences in phase" between the harmonic partials. Risset and Wessel (1999) point out that although this view was later put into perspective, the effects of phase differences between partials are quite weak and usually inaudible in reverberant rooms.

The authors mention a few objections against von Helmholtz's theory:

1. The fact that instruments are still recognisable even if a recording is heavily distorted (e. g. an old Schellack recording) or if transmission quality is very low (e. g. a telephone with limited frequency response) suggests that there must be a different factor that determines the timbre of an instrument than just the amplitudes of the harmonic partials. Likewise, the frequency response of a reverberant room can have large fluctuations depending on the position of the listener in the room and yet allows listeners to identify instruments.

2. The attack part of an instrument sound seems to be an important characteristic of the timbre. Omitting the attack part considerably reduces the recognition accuracy of human listeners (Stumpf, 1926).

3. The temporal evolution of the partials can also have an influence on the perceived instrument timbre. This can easily be demonstrated by a time-reversed piano sound which exhibits the same average partial spectrum as the original sound, but a clearly different timbre.

A considerable number of studies looked at the perceptual dimensions of musical timbre (e. g. Wedin and Goude, 1972; Miller and Carterette, 1975; Grey, 1977; Iverson and Krumhansl, 1993; McAdams et al., 1995) and tried to identify acoustic properties that correlate with these dimensions. Similarity ratings between instrument sounds were obtained from human subjects for a number of synthetic or recorded instrument samples. Multidimensional scaling techniques were then used to visualise the timbre space for a given number of dimensions. The target dimensions were correlated with acoustic features to determine perceptually relevant sound properties. Different authors report different properties to be important for the perception of the instrument timbre. Among these, purely spectral features such as descriptions of the spectral envelope (Wedin and Goude, 1972; Miller and Carterette, 1975; McAdams et al., 1995) as well as the spectral centroid (Iverson and Krumhansl, 1993) and the spectral energy distribution (Grey, 1977) appear to be among the main contributing factors. Some authors also report certain temporal features to correlate with the perceived timbre dimensions (Grey, 1977; Iverson and Krumhansl, 1993). Spectro-temporal features such as temporal fluctuations of the spectral envelope reported by some authors (Grey, 1977) have been rejected by other authors (McAdams et al., 1995; Caclin et al., 2005).

The above findings indicate that a large part of the timbre of an instrument can be captured by the average spectral envelope for each pitch. A computational timbre model should therefore accommodate information about prototypical sound spectra that capture the amplitude profiles of the different instruments in the recording. Since these amplitude profiles not only vary among instruments, but also for the pitches of each individual instrument, computational timbre models are required to capture this variation.

3.2 Non-negative analysis framework

Non-negative matrix factorisation (cf. Section 2.4) enables the extraction of prototype spectra (basis functions) from a recording and therefore provides a way to estimate timbre models of the underlying instruments in the recording. At the same time it enables us to utilise the prototype spectra for the detection of pitches over time.

3.2.1 Model

The following considerations are made regarding the non-negative analysis framework that is used for the remainder of this thesis:

- In order to be able to find temporal structures within a given spectrogram, we work with *single-frame spectral basis functions*. Musical note lengths are highly variable and for many musical instruments the player has a high degree of freedom to model the sound in various ways. This makes the use of extended basis functions with fixed temporal evolutions as in convolutive NMF (Section 2.4.3.1) unsuitable. Constraining the basis functions to a single frame has the advantage that all temporal information can be inferred from the non-negative gains.

- Previous studies (Sandell and Martens, 1995; Burred, 2008) confirmed in a PCA context that a representation of a note by a *single basis function* often explains more than 80% of the note's variance. A single basis function for each pitch might therefore be sufficient to unveil the occurrence of a note of a specific instrument within a magnitude spectrogram. In some cases, however, we might be interested in allocating *multiple* basis functions for notes of the same pitch in order to accommodate higher variations in the partial amplitudes, or to capture notes with different dynamics or playing styles (cf. Section 3.4).

- Within the course of a note, the pitch can vary if it is played or sung with vibrato or glissando. It is therefore indispensable to use prototype spectra with pitches on a sub-semitone resolution, that is, to have multiple basis functions for each nominal pitch of an instrument with pitch spacings of less than a semitone. A fine pitch resolution is also able to account for different temperament systems and tuning frequencies.

Based on these considerations, the non-negative analysis framework that is used throughout this thesis is formulated as

$$\mathbf{V} \approx \mathbf{\Lambda} = \sum_{i=0}^{I-1} \sum_{\phi=0}^{\Phi-1} \overset{\phi\downarrow}{\mathbf{W}}^{\phi,i} \mathbf{H}^{\phi,i}. \tag{3.1}$$

The matrix $\mathbf{V} \in \mathcal{R}_+^{K \times N}$ with K frequency bins and N time frames represents the constant-Q magnitude spectrogram with a logarithmically-spaced frequency axis. It is approximated by $\mathbf{\Lambda} \in \mathcal{R}_+^{K \times N}$ which has the same dimensions as $\mathbf{V}$ and consists of a superposition of spectra for several instruments and pitches. $\mathbf{W}^{\phi,i} \in \mathcal{R}_+^{K \times R}$ contains the basis functions for all I instruments and Φ pitches, and $\mathbf{H}^{\phi,i} \in \mathcal{R}_+^{R \times N}$ is the corresponding gain matrix. R denotes the specified number of spectral templates for each pitch of each instrument. The operator $\phi\downarrow$ denotes a downward shift of the matrix elements by ϕ rows while the upper ϕ rows are filled with zeros as in Section 2.4.3.2.

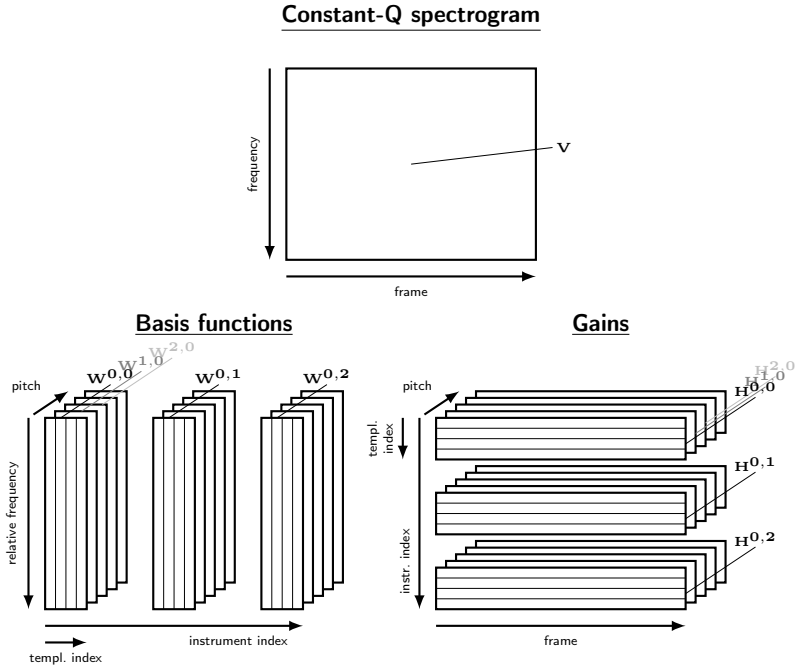


Figure 3.1: Graphical illustration of the non-negative analysis framework.

A schematic illustration of the framework is displayed in Fig. 3.1. In this model, each instrument is represented by a 3-dimensional structure (tensor) that contains a fixed number of basis functions for each pitch. The pitch resolution is determined by the frequency resolution of the constant-Q analysis spectrogram and the number of templates per pitch can be chosen arbitrarily. Likewise, for each instrument a 3-dimensional structure contains the corresponding gains for the spectral templates. The concatenation of all gains with the same instrument and template index provides a matrix that contains the gain trajectories of a specific template. This can be seen as a *pitch activation function* for that template and corresponds to a horizontal layer in the gain structure of Fig. 3.1. In order to arrive at a single piano roll-like representation for each instrument, the pitch activation function for all templates belonging to the same instrument can be summed, which corresponds to a sum along the vertical direction in a gain tensor in Fig. 3.1.

In this model, all templates in $\mathbf{W}^{\phi,i}$ are aligned and have their first partial at the first row index. Due to the use of the logarithmic frequency axis, the remaining harmonic partials of all basis functions likewise appear roughly at the same row index. This alignment has advantages for the estimation of missing spectral templates in Chapter 4. The mixture model shifts each basis function to the correct frequency position.

The non-negative analysis framework in Eq. (3.1) is not very different from the basic NMF algorithm (see Section 2.4.1). It mainly imposes a structure on the basis functions and gain matrices that provides an intuitive representation of the underlying instruments and enables the examination of transcription results through a piano roll-like gain structure. It also allows us to incorporate prior knowledge by initialising either the basis functions or the gain matrices in different ways as will be outlined in Section 3.3.

3.2.2 Update equations

The update equations for both $\mathbf{W}^{\phi,i}$ and $\mathbf{H}^{\phi,i}$ based on the β-divergence between $\mathbf{V}$ and $\mathbf{\Lambda}$ are given by

$$\mathbf{W}^{\phi,i} \leftarrow \mathbf{W}^{\phi,i} \bullet \frac{\left(\overset{\phi\uparrow}{\mathbf{V}} \bullet \mathbf{\Lambda}^{\overset{\phi\uparrow}{\beta-2}}\right) \left[\mathbf{H}^{\phi,i}\right]^{\top}}{\left(\mathbf{\Lambda}^{\overset{\phi\uparrow}{\beta-1}}\right) \left[\mathbf{H}^{\phi,i}\right]^{\top}} \tag{3.2}$$

$$\mathbf{H}^{\phi,i} \leftarrow \mathbf{H}^{\phi,i} \bullet \frac{\left[\overset{\phi\downarrow}{\mathbf{W}^{\phi,i}}\right]^{\top} \left(\mathbf{V} \bullet \mathbf{\Lambda}^{\beta-2}\right)}{\left[\overset{\phi\downarrow}{\mathbf{W}^{\phi,i}}\right]^{\top} \mathbf{\Lambda}^{\beta-1}}. \tag{3.3}$$

In these equations, $\bullet$ denotes an elementwise multiplication and all divisions and exponentiations are likewise carried out on a per-element basis. $\top$ is used as the symbol for matrix transposition. The derivation of the update equations can be found in Appendix A. An implementation is available from `http://code.soundsoftware.ac.uk/projects/svnmd`.

The model in Eq. (3.1) is not a low-rank approximation of the spectrogram, and therefore the application of the update equations in a completely unsupervised manner will not yield useful results for both the basis functions and the gain structure. The model is highly underdetermined as the number of parameters to estimate in the model is larger than the number of elements in the input spectrogram. A useful estimation can only be achieved when a certain amount of prior information is provided, which is exactly what is aimed for in a semi-automatic transcription system.

3.2.3 Evaluation metrics

Different metrics have been used for the evaluation of multiple-f0 estimation and note tracking systems. Most of these metrics rely on comparisons between the detected pitches of a transcription algorithm and those pitches that are actually

present in the recording. The metrics for the MIREX evaluation on frame-level[1] are based on precision and recall measures. Another set of metrics was proposed by Poliner and Ellis (2007a). It includes error scores for note substitutions, missing notes, and false positives.

These measures, however, are not so well suited for the evaluation of the analysis results of the non-negative framework in Section 3.2 for two reasons: First, the common transcription metrics are not designed for multi-instrument transcriptions which require the evaluation of the instrument assignments of the detected notes. And second, these measures only evaluate the performance of a complete transcription system without any insights into the contributions of the individual processing stages.

A prevalent and recurring concept in many transcription systems is the use of a *pitch activation function* that indicates the likelihood of pitches over time on a continuous scale. Based on these functions, decisions about active pitches are made, either by defining thresholds or by more advanced detection mechanisms. An evaluation of transcription systems based on these decisions will thus necessarily evaluate both the pitch activation function *and* the decision process.

In the following paragraphs, a set of error measures is presented that is capable of evaluating the quality of an activation function as well being able to deal with parts-based transcriptions. In our case and also more generally for NMF algorithms, gain matrices can be seen as pitch activation functions which aim at high values at time-frequency positions of active notes and low values where notes are absent. We express the pitch activation function of an instrument i by $\mathbf{G}^i$, where the matrix components are defined as

$$\left[\mathbf{G}^i\right]_{\phi,n} = \sum_{r=0}^{R-1} \left[H^{\phi,i}\right]_{r,n} . \tag{3.4}$$

In the same way as above, ϕ denotes the pitch index, n the time index, and R corresponds to the number of templates per instrument and pitch. $\mathbf{G}^i$ corresponds to a non-binary piano roll representation of the pitches associated with instrument i.

[1] `http://www.music-ir.org/mirex/wiki/2011:Multiple_Fundamental_Frequency_Estimation_&_Tracking#Evaluation`

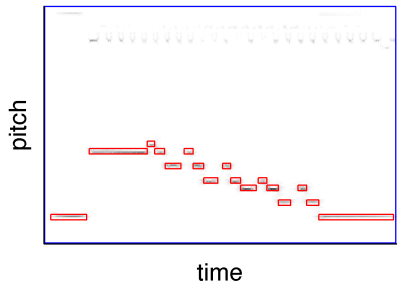


Figure 3.2: Graphical illustration of the pitch precision measure PP_i. The diagram displays a gain matrix $\mathbf{G}^i$. The red boxes highlight the ground truth time-pitch bins which are considered in the numerator of Eq. (3.5). The blue box contains all bins in this gain matrix over which the sum in the denominator of Eq. (3.5) is computed. Activations at the correct pitches are visible as well as some spurious activations (in the upper part of the matrix).

Pitch precision

The *pitch precision* measure computes for each instrument i the amount of energy in the gain matrix $\mathbf{G}^i$ that is concentrated in the ground truth fundamental frequencies and relates it to the overall energy in the matrix:

$$\mathrm{PP}_i = \frac{\sum_{n=0}^{N-1} \sum_{\phi \in \mathcal{F}_{n,i}} \left(\left[\mathbf{G}^i \right]_{\phi,n} \right)^2}{\sum_{n=0}^{N-1} \sum_{\phi'=0}^{\Phi-1} \left(\left[\mathbf{G}^i \right]_{\phi',n} \right)^2}. \tag{3.5}$$

In this equation, $\mathcal{F}_{n,i}$ denotes the set of annotated ground truth pitches in the n-th time frame for instrument i.

This measure is graphically illustrated in Fig. 3.2 for an example gain matrix $\mathbf{G}^i$. The red boxes highlight the ground truth time-pitch bins that are considered in the numerator, and the blue box indicates the bins over which the sum in the denominator is computed. For monophonic instruments, $\mathcal{F}_{n,i}$ contains at most one note at each time frame. In an ideal scenario, all energy would be concentrated in the fundamental frequencies, which would enable an accurate detection of notes with the pitch activation function $\mathbf{G}^i$. This case would correspond to a pitch precision of $\mathrm{PP}_i = 1$.

The name *pitch precision* is used here because this measure is very similar to the precision measure in a classification context, which is defined as the ratio

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}. \tag{3.6}$$

Instrument precision

Similar to the pitch precision, the *instrument precision* computes the amount of energy concentrated in the ground truth fundamental frequencies of an instrument and relates it to the overall amount of energy at the same fundamental frequencies across all instruments:

$$\mathrm{IP}_i = \frac{\sum_{n=0}^{N-1} \sum_{\phi \in \mathcal{F}_{n,i}} \left(\left[\mathbf{G}^i \right]_{\phi,n} \right)^2}{\sum_{n=0}^{N-1} \sum_{\phi \in \mathcal{F}_{n,i}} \sum_{i'=0}^{I-1} \left(\left[\mathbf{G}^{i'} \right]_{\phi,n} \right)^2}. \tag{3.7}$$

A graphical illustration can be found in Fig. 3.3. In the upper diagram, the gain matrix $\mathbf{G}^0$ for the first instrument in a 2-instrument mixture is shown and the ground truth time-pitch bins are highlighted in red. The numerator in Eq. (3.7) computes the sum over the squared values contained within these bins. The lower two diagrams of this figure show the gain matrices $\mathbf{G}^0$ and $\mathbf{G}^1$ of the two instruments. The sum in the denominator in Eq. (3.7) is computed over the time-pitch bins indicated by the blue boxes.

The instrument precision measure is thus a measure for the separability of the instruments based on the given pitch activation functions of all instruments. Ideally, all energy should be concentrated in the one instrument that actually produced the current note in the recording. This would prevent the algorithm from assigning the note to the wrong instrument and would result in an instrument precision IP_i of 1.

Combined precision

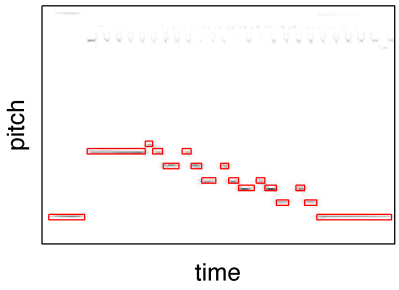
Pitch precision and instrument precision are here joined into a *combined precision* measure which is computed as the harmonic mean of the two metrics:

$$\mathrm{CP}_i = 2 \cdot \frac{\mathrm{PP}_i \cdot \mathrm{IP}_i}{\mathrm{PP}_i + \mathrm{IP}_i}. \tag{3.8}$$

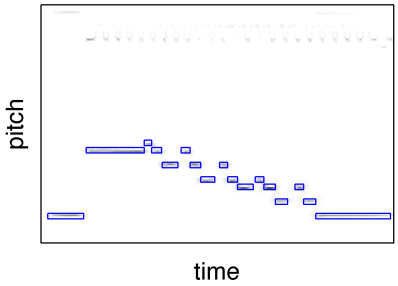
The combined precision results in a value of 1 when both pitch and instrument precision take on a value of 1 and it converges toward 0 when both pitch and instrument precision decrease.

3.3 Generic templates vs specific templates

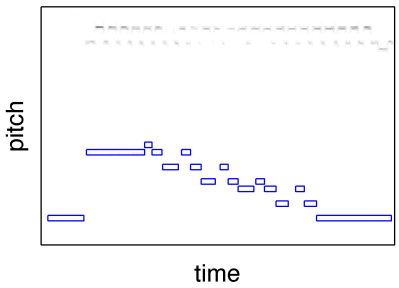
In this thesis, a *timbre model* of an instrument is given by a complete set of basis functions across the whole pitch range, and can contain multiple templates



(a) Gain matrix $\mathbf{G}^0$ of the first instrument.



(b) Gain matrix $\mathbf{G}^0$ of the first instrument.

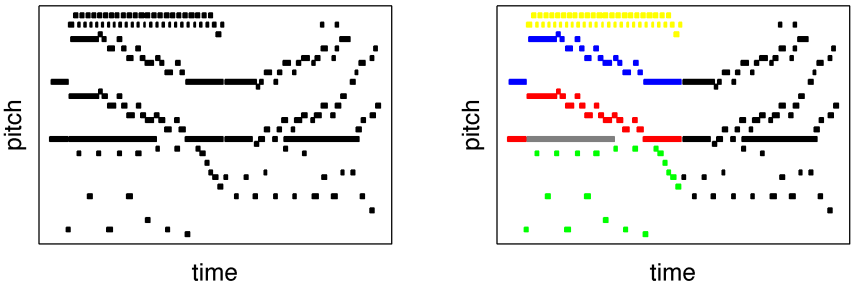


(c) Gain matrix $\mathbf{G}^1$ of the second instrument.

Figure 3.3: Graphical illustration of the instrument precision measure IP_0 in a 2-instrument mixture. (a) displays the gain matrix $\mathbf{G}^0$ for the first instrument and highlights the bins over which the sum in the nominator of Eq. (3.7) is calculated. (b) and (c) display the gain matrices $\mathbf{G}^0$ and $\mathbf{G}^1$ of both instruments. The blue boxes indicate the bins that are considered in the denominator of Eq. (3.7).

per pitch. The timbre model of a single instrument i_0 is thus given by the set of basis functions $\mathbf{W}^{\phi,i_0}$ for all ϕ in Eq. (3.1). When appropriate timbre models are available for all instruments in the recording under analysis, the gain matrices can be inferred by applying the update in Eq. (3.3) for a certain number of iterations.

We explore two ways in which timbre models can be established in the non-negative analysis framework: Either the basis function matrices $\mathbf{W}^{\phi,i}$ are set to pre-extracted instrument spectra of the instrument types in the recording, or alternatively basis functions can be directly extracted from the recording. In the first case, spectra would have to be learned from a database of instrument sounds. Each way requires different types of user information. In the first case, the user provides the *instrument identities* of the instruments contained in the recording under analysis, so that the corresponding sets of spectra can be selected. In the second case, the user provides information that enables the extraction of spectra of the instruments in the recording. This could be achieved by asking the user



(a) (b)

Figure 3.4: Prototype user-interface for note labelling. The user is provided with a piano roll of a fully-automatic transcription algorithm (a) and asked to assign notes to the instruments by colouring individual note objects (b).

to provide information about onset and offset times of notes at different pitches for each instrument in the recording.

Providing note information for each instrument might seem like a laborious process that violates the third criterion in Section 2.3.2. This process, however, can in practice be greatly facilitated by providing the user with an initial piano roll representation obtained by a fully-automatic transcription system and asking the user to assign detected note objects to the different instruments. An interface could be designed that represents the instruments by different colours and allows a user to assign the notes by changing their colour to that of the corresponding instrument. An example of such a prototype interface can be seen in Fig. 3.4. We will refer to this process as the *note labelling* approach.

A good overview of research on the human ability to identify musical instruments was provided by Martin (1999, Sect. 3.1). Most studies on this subject focused on isolated notes without considering the larger musical context such as note sequences or melodic phrases. The results suggest that the identification accuracy depends on various factors such as the number of instruments tested, the instrument type, and the musical background of the subjects. Absolute accuracies vary considerably among the different studies, but many studies report accuracies above 80 % particulary when instruments of different instrument families were tested (e. g. Strong and Clark, 1967; Berger, 1964; Kendall, 1986). Martin (1999) points out that identification accuracies can be expected to be even higher when musical context is taken into account (e. g. using musical phrases rather than isolated notes), as confirmed by Kendall (1986). Kendall and Carterette (1993) also investigated the identification of instruments in multi-instrument settings and found that the identification is strongly dependent on the blend of the instrument timbres: accuracies were lower for instrument com-

binations that blended well and vice versa. One aspect that is not considered in this study, though, is prior knowledge about typical instrument combinations of ensembles: when listening to a 4-instrument string ensemble for example, experienced listeners might be able to identify the ensemble as a string quartet and to name its constituent instruments based on their knowledge.

The ability to assign individual notes to the underlying instruments depends on the user's ability to read music. Following the score while listening to a performance of the same piece requires a mapping of perceived events to note symbols in the score. Obviously, music reading skills are acquired through musical education and hence readers without musical training cannot be expected to be able to reliably follow a score. Research on music reading has on the one hand focused on processes involved when sight-reading a piece of music by analysing eye movement patterns (e. g. Goolsby, 1989) or briefly exposing participants to chunks of notated music in order to investigate the amount of information that can be memorised (Sloboda, 1984). More relevant to the investigated type of user information is the ability of readers to passively follow a performance rather than actively reproduce the notated music. Studies on error detection in score reading are hence of interest as they can provide evidence of how accurately a reader is able to connect performance and notation. Studies by Hansen (1961) and Gonzo (1971) found that participants with a background in music theory as well as those with piano skills achieved better error detection results than participants without these preconditions. Both studies were based on choral music with a limited number of voices. It can hence be assumed that musicians are — to varying degrees — able to follow the individual voices of a music performance and hence to assign individual notes to the underlying instruments.

Experiments were carried out to compare these two different types of user information, *providing instrument identities* and *note labelling*, which are explained in the following sections. Timbre information was extracted based on the user information and these timbre models were employed to compute pitch activation functions for a dataset of instrument mixtures with varying polyphonies. Quality measurements of the pitch activation functions provide insights into the suitability of the user information for the transcription task.

In the following Section 3.3.1, the process of extracting timbre information from a recording is illustrated. The computation of the gain matrix is described in Section 3.3.2. The evaluation procedure including a description of the dataset and the results of the experiment is explained in Section 3.3.3.

3.3.1 Learning the basis functions

Knowledge about the instrument identities in the recording under analysis allows us to employ previously extracted timbre models of the same instrument types from a musical instrument database. For the experiment introduced above, timbre models were learned from the RWC musical instrument database (Goto et al., 2002) for each instrument identified by the user as being present in the target (test) recording. The RWC database consists of audio files of monophonic recordings of instruments playing a chromatic scale over their whole compass recorded in a dry studio environment. Each note has a duration of a few seconds and in some cases more than one note is played per pitch (for bowed string instruments for example, notes with the same pitch are played on different strings). These recordings were manually annotated and the annotations were stored in MIDI format. To learn the basis function sets from this training data, the update rule described in Eq. (3.2) was used, fixing the gains $\mathbf{H}^{\phi,i}$ to contain ones at the frequency bins and time frames corresponding to the notes in the training data annotation and zeros elsewhere. The exact frequency bins were determined by finding the maximum within the frequency region of a semitone in the corresponding constant-Q spectrogram of each note at each time frame. The number of templates R was set to 1. The constant-Q analysis covered the frequency range from C2 ($\sim 65\,\mathrm{Hz}$) to C8 ($\sim 4.2\,\mathrm{kHz}$) with a frequency resolution of 48 bins per octave and was computed by means of a MATLAB toolbox (Schörkhuber and Klapuri, 2010). This implementation determines the hop size as a function of the length of the shortest analysis window. In our experiments the time resolution was given by 4.1 ms. Matrices $\mathbf{W}^{\phi,i}$ were randomly initialised and 10 iterations of the update function were computed which provided reasonable convergence results in practice.

For the second type of user information, the labelling of notes for each instrument in the target mixture, a similar learning procedure was applied to derive the basis functions directly from the target data. For each target mixture, a set of basis functions was learned for each instrument from the *polyphonic* target mixture itself. The gain matrices were again initialised by ones at the time-frequency positions of the user annotations and zeros elsewhere. All basis functions were learned jointly by applying the update rule for $\mathbf{W}^{\phi,i}$ in Eq. (3.2) for 10 iterations. Again only $R = 1$ template was learned per instrument and pitch.

We here assume that the user has labelled *all* notes of *all* instruments which provides an upper performance limit for this type of user information. In a practical application for user-assisted transcription, the user would only be required to label a small number of notes for each instrument. An investigation

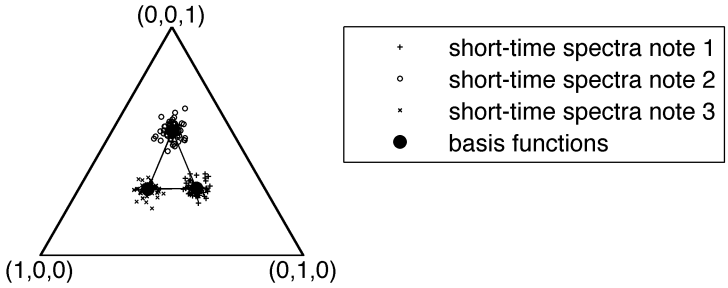


Figure 3.5: Illustration of the effect of note-labelling. The note labels provided by a user assign short-time spectra to the same cluster and the basis functions are given by the mean of these spectra.

of the effect of having a reduced set of annotated notes will be discussed in Chapter 4.

The effect of note-labelling is graphically illustrated in Fig. 3.5 in the same simplex diagram that was used in Section 2.4.2. It shows data sampled from three i.i.d. Gaussian distributions with different means. By labelling a note, the user assigns the spectra at the different time frames to the same basis function. In the simplex diagram this corresponds to assigning the data points to the same cluster, indicated by the different markers in Fig. 3.5. Keeping these cluster assignments fixed, NMF will find a basis function that minimises the reconstruction error for the assigned spectra. This basis function is given by the mean of the assigned spectra. Note that the mean does not necessarily correspond to the *arithmetic mean* as it depends on the applied cost function.

Both the individual instrument scales of the RWC dataset as well as the phrases of the test dataset were recorded under similar conditions in a relatively dry studio environment. The RWC database consist of single note recordingssingle record

3.3.2 Learning the gains

The two types of timbre models were employed for the extraction of pitch information from the spectrogram. In a practical application of semi-automatic transcription, no prior information about the gains of each spectrum is available. The gain matrices $\mathbf{H}^{\phi,i}$ were initialised with absolute values of Gaussian noise with a variance of 1. The gains were learned by initialising the matrices $\mathbf{W}^{\phi,i}$ with one of the basis function sets described in Section 3.3.1, and the update

rule for the gain matrices $\mathbf{H}^{\phi,i}$ (Eq. (3.3)) was applied while keeping the basis functions fixed. Again, a fixed number of 10 iterations was computed.

3.3.3 Evaluation

3.3.3.1 Dataset

A test set was constructed based on monophonic musical phrases from 12 different acoustical instruments. The instrument types were: flute, oboe, clarinet, bassoon, alto sax, horn, trumpet, trombone, tuba, violin, viola and violoncello. Most of the phrases were recorded and kindly provided by Martin (1999), the remaining recordings stem from the author's personal collection. All phrases were recorded in a studio environment without significant amount of reverb. Each of the signals had a length of approximately 30 s. From these phrases, random mixtures of 2, 3, 4 and 5 instruments were generated by summing the amplitude-normalised signals. 50 mixtures were generated for each polyphony level.

For each monophonic phrase, pitch, onset time and offset time of each note were manually annotated and stored as MIDI files. The ground truth for each instrument mixture was created by combining the ground truth annotations of the instruments contained in the mixture.

3.3.3.2 Results

The results of the experiments are displayed in Fig. 3.6. From left to right, the results for the different numbers of instruments are displayed. Each row shows the results for the metrics introduced in Section 3.2.3: *pitch precision* (PP_i), *instrument precision* (IP_i) and *combined precision* (CP_i). Within each panel, the results for the two different types of timbre models can be compared: 'data' identifies the timbre models learned from the mixtures themselves whereas 'RWC' denotes the timbre models based on the generic instrument spectra from the RWC database. Both types of timbre models were each evaluated with the two cost functions *least squares error (LS, $\beta = 2$)* and the *Kullback-Leibler divergence (KL, $\beta = 1$)* (cf. Section 2.4.1). The results are displayed as box plots. Each box plot summarises the results for all individual instruments in all mixtures of the same polyphony. The upper and lower edges of the box represent the *first* (Q_1) and *third quartile* (Q_3), the *median* is displayed in between. The whiskers extend to the *5% quantiles* and all points outside this interval are marked by crosses and considered as *outliers*.

Overall, the results for both the pitch precision PP_i and instrument precision IP_i show very similar tendencies and hence the combination of these two, the combined precision CP_i, exhibits that same tendency. A comparison of the

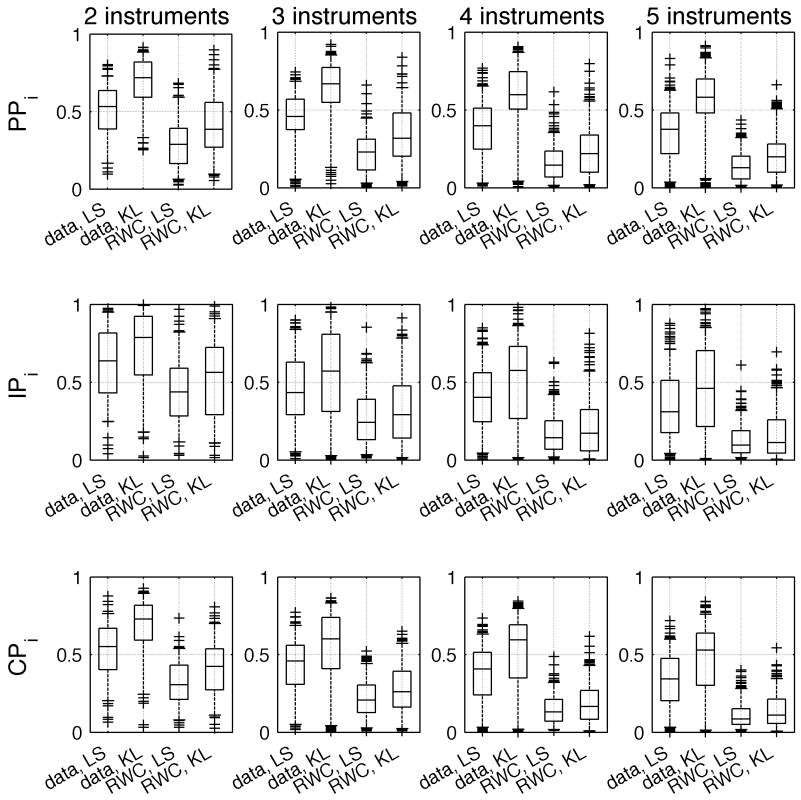


Figure 3.6: Evaluation results for the comparison between the two different timbre models. Panels from left to right show the results for the different polyphony levels. From top to bottom the three metrics *pitch precision* (PP_i), *instrument precision* (IP_i) and *combined precision* (CP_i) are illustrated. In each panel, the two different timbre models 'data' and 'RWC' in combination with the cost functions 'KL' and 'LS' can be compared. In each box plot, the median of the results is indicated in the middle of the box, the edges mark the lower and upper quartile, and the whiskers extend to the minimum and maximum data points.

different *cost functions* exhibits that in all cases the KL-divergence leads to more accurate results and is therefore a better choice than the LS cost function. This is consistent across all polyphony levels, i. e. from 2 instruments to 5 instruments, and for all metrics.

Comparing the different timbre model types makes it obvious that basis functions learned from the recordings under analysis themselves (data) lead to considerably better results than basis functions learned from the RWC database (RWC). The use of timbre models learned from the mixtures not only leads to more accurate gain matrices for the *individual instruments*, which is captured by PP_i, it also limits *instrument confusions*, that is, false assignments of the gains of one instrument to another instrument, which is captured by IP_i. This tendency can be observed for all polyphony levels. The relative differences in accuracy between the timbre models for the KL divergence become larger when the number of instruments increases: the median of the combined precisions for 'data, KL' and 'RWC, KL' amount to 73% and 42% for 2-instrument mixtures, for 5-instrument mixtures these measures amount to 52% and 11%, respectively.

The findings of this first experiment confirm that user information about individual notes of each instrument can provide timbre models that obtain significantly more accurate gain matrices than using timbre models consisting of generic note templates for the instruments in the recording. Obviously in this experiment, all notes of all instruments have been employed for the estimation of the timbre models, so that these results have to be interpreted as an upper limit for the accuracy that can be achieved by this timbre model, rather than results achieved in practice. In a more realistic scenario, accuracies will most likely be lower, and we will adress this issue in Chapter 4. Nevertheless these results encourage further investigations of this type of user information. In the following section, a method is proposed that allows us to extract more refined timbre models with several spectral templates per instrument and pitch.

3.4 Single templates vs multiple templates per instrument and pitch

Based on the results presented in Section 3.3, the use of templates extracted from the recording under analysis was further investigated. In the previous experiment, a timbre model characterised the pitch of an instrument by a *single spectral template*, that is, a single prototype spectrum. Even though this model might be a reasonable first approximation of the short-time spectra in successive time frames, the spectral shapes of these short-time spectra are usually not static and the amplitudes of the harmonics for a given pitch often vary non-uniformly

over time. In addition to these variations in the steady-state part of a note, different dynamics and playing styles can likewise cause variations in the shape of a short-time analysis spectrum of notes for a given instrument and pitch.

Figure 3.7 shows examples of various spectral shapes for different instruments. All spectra are normalised so that their elements sum to one. The three panels on the left hand side display short-time spectra of the steady-state part of notes with the same pitch played by a trumpet at three different dynamic levels: piano ($\boldsymbol{p}$), mezzoforte ($\boldsymbol{mf}$), and forte ($\boldsymbol{f}$). Spectra at higher dynamic levels clearly exhibit more energy in the higher partials, while the energy of the fundamental is decreased. The panels in the centre of Fig. 3.7 illustrate spectra of different playing styles of a violin: normal bowed playing style (normal), bowed with mute (con sord.), and plucked (pizz.). In the spectrum of the muted sound (con sord.), harmonics 5 and 6 are clearly damped compared to the normal playing style, and the plucked spectrum (pizz.) contains considerably less energy in the upper partials and additional frequency content at the lower end of the spectrum. The panels on the right hand side show spectra at the beginning, middle and end of the same piano note. Partials decay at different rates which causes variations in the frequency content. These examples illustrate the dynamic nature of the instrument spectra and the fact that spectra of the same instrument at the same pitch can vary considerably. This motivates the use of timbre models that can account for such spectral variations.

In this section a method is proposed that extracts *multiple* spectral templates for each instrument and pitch based on the note labels provided by a human user (cf. Section 3.3). A novel learning algorithm based on k-means clustering is proposed that infers $R > 1$ templates for each pitch of each instrument within the non-negative analysis framework presented in Section 3.2. In the following section, the individual steps of the learning algorithm are explained and graphically illustrated. In Section 3.4.2 the method is experimentally evaluated.

3.4.1 Learning the basis functions

In order to learn multiple templates per instrument and pitch, it is *not* possible to apply the same method as in Section 3.3.1. Setting the gains of all R templates at the time frames of an annotated note of an instrument to ones, and applying the update functions for the spectral templates (Eq. (3.2)), can lead to undesirable results. This method would enable the NMF algorithm to assign different parts of the spectrum to different templates and thus allows the partials of a note to be split among the different templates. A template that only contains a subset of partials, however, might be used by the algorithm to explain partials of other notes from the same or another instrument. An intuitive example for this case

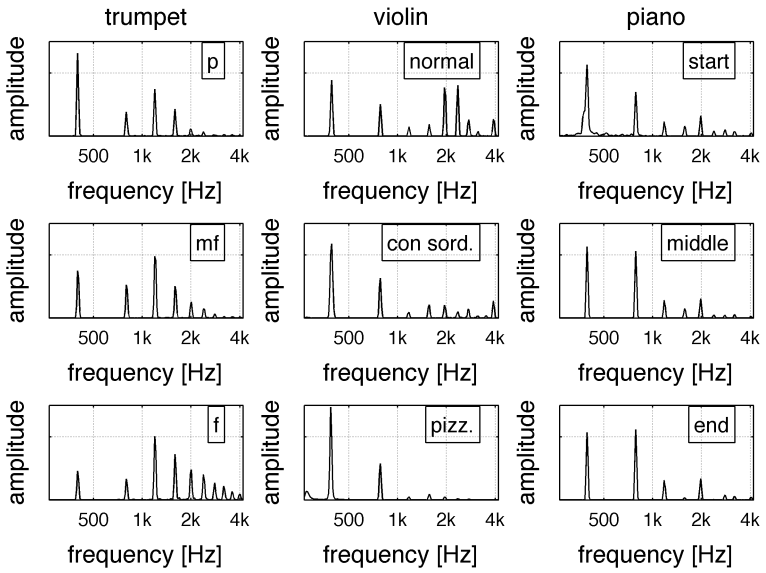


Figure 3.7: Varying short-time spectra of different instruments. Left: trumpet spectra with different dynamics, centre: violin spectra with different playing styles, right: piano spectra at different parts of a note.

would be a spectral template that only contains a single partial (i. e. a single spectral peak) which can be used by the algorithm to approximate a partial of any note at that frequency position of the same or another instrument. This would produce a gain value either at the wrong fundamental frequency or the wrong instrument or both and thereby adulterate the transcription accuracy. Instead, we would like each template to summarise a certain *subset of the short time spectra*, so that it can be used to detect short time spectra with a similar shape in the recording.

Assuming that a user has labelled note onset, note offset and pitch of notes played by the instruments in the recording at several pitches, an algorithm is proposed that extracts a fixed number of spectral templates for each annotated pitch of each instrument. In this section, the procedure for learning a fixed number of templates is illustrated for a single user-annotated note.

Figure 3.8 illustrates the iterative process of learning R templates for a single note labelled by the user. The user provides information about the start frame, the end frame and the pitch ϕ_0 of a note of a particular instrument i_0. This information can be illustrated by a piano roll that contains a single line representing the note, as shown on the left-hand side of Fig. 3.8a. Given this information, we can identify the matrix $\mathbf{W}^{\phi_0,i_0}$ in which the learned templates will be stored and the matrix $\mathbf{H}^{\phi_0,i_0}$ that contains the gains for each of the

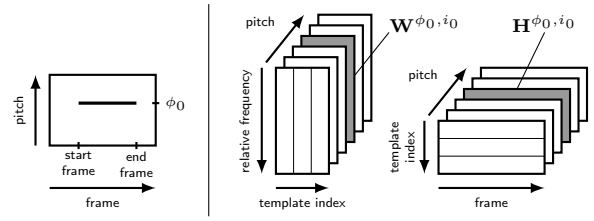
templates over time (grey-shaded matrices on the right-hand side of Fig. 3.8a). Since only those two matrices $\mathbf{W}^{\phi_0,i_0}$ and $\mathbf{H}^{\phi_0,i_0}$ are relevant for learning the templates from the labelled note, we isolate them from their tensors when illustrating the learning algorithm in Fig. 3.8b–f.

In Fig. 3.8, panels b–f display the algorithmic steps for estimating the spectral templates. This procedure is in fact very similar to applying *k-means clustering* to the spectra of a note at all time frames within the spectrogram $\mathbf{V}$. In this analogy, each spectral template corresponds to a cluster mean and thus represents a set of spectra at different time frames. Since the learning procedure is carried out within the non-negative framework, the correponding k-means clustering steps might not be obvious. For that reason, we illustrate these on the right hand side of panels b–f. In these graphs, each data point corresponds to a short-time spectrum of the note at a particular time frame in the K-dimensional space which is here reduced to 2 dimensions for the sake of illustration.

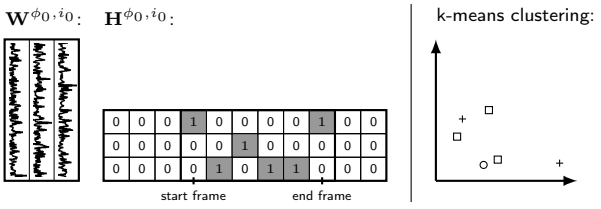
1. **Initialisation:** The algorithm starts by initialising the spectral templates in $\mathbf{W}^{\phi_0,i_0}$ with non-negative random values (Fig. 3.8b). In the gain matrix $\mathbf{H}^{\phi_0,i_0}$, each frame of the note is randomly assigned to exactly one spectral template by setting the corresponding gains to a value of 1 while all other entries of the matrix are set to 0. In the k-means example, this corresponds to assigning the data points randomly to one of the three clusters, depicted by crosses, circles and squares.

2. **Update:** In the second step (Fig. 3.8c), the spectral templates in $\mathbf{W}^{\phi_0,i_0}$ are updated according to Eq. (3.2) based on the gains that were set in the previous step. This modifies the spectral templates in such a way that the resulting templates minimise the β-divergence at the assigned frames. Thus, each resulting spectral template can be seen as an average of the instrument spectra at the time frames that were assigned to it. In k-means clustering terms, this is equivalent to computing the average of the data points that were assigned to the same class. Note that in order to eliminate scale-ambiguities in the non-negative framework, all spectral templates in $\mathbf{W}^{\phi_0,i_0}$ are scaled to have a power of 1 and the gains are adjusted accordingly.

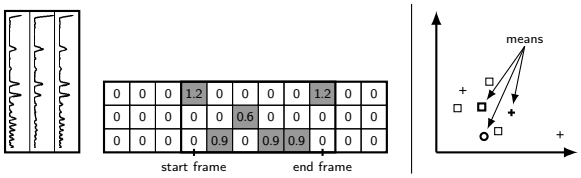
3. **Assignment:** In order to re-assign the note spectra at all time frames to the template that best resembles their spectral shape, the template gains at each note frame are set to equal values (Fig. 3.8d) and the gains are updated based on the given spectral templates (Fig. 3.8e) according to Eq. (3.3). This way, the gain matrix contains the contributions of each template to the audio spectra of each time frame when linearly combining



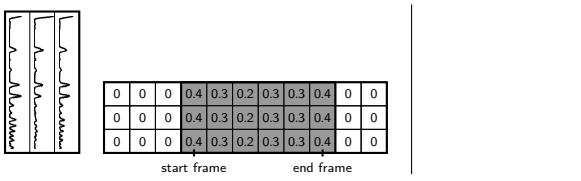
(a) Piano roll and non-negative analysis framework



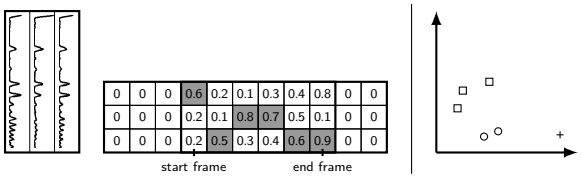
(b) Initialisation



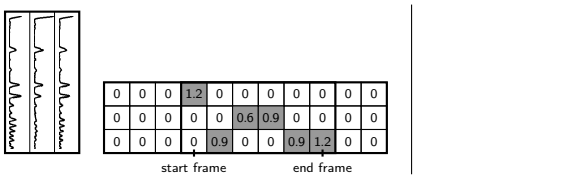
(c) Update



(d) Assignment (1)



(e) Assignment (2)



(f) Assignment (3)

Figure 3.8: Learning multiple spectral templates based on a single user-annotated note.

the templates. This can be seen as a similarity measure between the templates and the spectra. We assign each frame to the template with the highest gain value, here indicated by the grey-shaded entries. In the k-means clustering example, this corresponds to the assignment step, in which each data point is assigned to the closest mean. A new matrix $\mathbf{H}^{\phi_0,i_0}$ is set up (Fig. 3.8f) that contains at each frame and each assigned template index the gains from step 2 (cf. Fig. 3.8c), and zeros elsewhere.

The algorithm iterates over steps 2 and 3.

The reason for assigning each frame to just a single spectral template in steps 1 and 3 is exactly the same as described in the beginning of this section: we would like each template to represent a subset of the short time spectra of the note. Assigning the frames to multiple templates would enable the algorithm to explain different spectral parts by different templates.

In k-means clustering, there is a chance of producing empty clusters when assigning the data points to the new means. The same problem applies to our proposed learning algorithm. In our algorithm this problem can occur in Fig. 3.8e, when for a certain template none of the frames contains the largest gains. In this case, we detect the largest cluster (i. e. the template with the largest number of assigned frames) and randomly assign half of its frames to the empty cluster. The spectral template of the empty cluster is thereby discarded.

Although the learning procedure was here illustrated by an individual note of a single instrument, the procedure is applicable to and intended for polyphonic audio. A MATLAB implementation of the learning algorithm is available at `http://code.soundsoftware.ac.uk/projects/svnmdmt`.

3.4.2 Evaluation

The evaluation of the proposed template learning algorithm was carried out in two experiments. In the first experiment (Section 3.4.2.1) the upper limit of performance of the algorithm was explored when used for semi-automatic transcription. The results of this experiment provide some intuition about the potential of the framework to accurately approximate a spectrogram. The second experiment (Section 3.4.2.2) looked at a more realistic semi-automatic transcription setting in which only a part of the notes are employed for learning the templates which are then applied to transcribe the remainder of the recording.

3.4.2.1 Experiment 1: Exploring the upper performance limit

In the first experiment we explored the upper performance limit of the template learning algorithm when applied to semi-automatic music transcription. This

upper bound is given when a user labels *all* notes of *all* instruments in the mixture under analysis. Although this scenario may seem trivial, because no transcription algorithm would be required if all notes were known beforehand, this evaluation provides an intuition about the expressivity of the algorithm and reveals any methodological flaws.

Experimental setup

For each file in the dataset, we extracted $R = 1$, 3 and 5 templates per pitch, by applying the template learning algorithm described in Section 3.4.1. We ran 50 iterations of the learning algorithm in order to ensure good convergence. The user information was given by the ground truth MIDI files of the instruments contained in the mixture which contained onset, offset and pitch information of the notes of the instruments. Once the basis functions were learned from the constant-Q magnitude spectrogram of the recording, the gain matrices were computed. This was done by randomly initialising all matrices $\mathbf{H}^{\phi,i}$ with non-negative values and applying 10 iterations of the update equation for the gains (Eq. (3.3)). The transcription metrics described in Section 3.2.3 were employed. The experiment was conducted for the KL-divergence ($\beta = 1$) and the IS-divergence ($\beta = 0$).

Results

The results of this experiment are displayed in Fig. 3.9. The upper panels display the results obtained by using the Itakura-Saito (IS) divergence, the lower panels the results of the Kullback-Leibler (KL) divergence. From left to right, the panels show the results of the different polyphony levels from 2 to 5 instruments. In each panel, the combined precisions CP_i of all instruments of all files represented by different numbers of templates per pitch can be compared.

In this experiment, the Itakura-Saito divergence consistently achieves higher combined precisions than the Kullback-Leibler divergence for all polyphony levels and all numbers of templates. A possible explanation for the good performance of the IS-divergence is its scale-invariance property (Févotte et al., 2009) which is in compliance with the Weber-Fechner law (Fechner, 1860) applied to the perception of loudness as indicated in Section 2.4.1. When comparing the results for different numbers of spectral templates per instrument and pitch, a slight but consistent increase of IP_i can be observed when more templates are learned for each note. For the IS-divergence (upper row) the median increases by 4–7 percentage points when 5 templates are used as opposed to just 1 template. For the KL-divergence this increase is in the range of 0.5–5 percentage points. The results show that learning multiple templates per instrument and pitch according

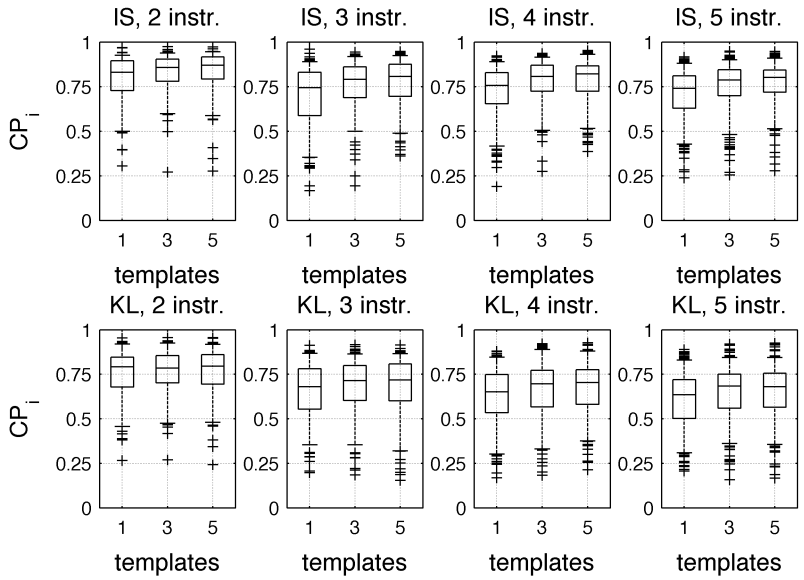


Figure 3.9: Evaluation results for experiment 1. Panels from left to right show the results for the different polyphony levels. In the upper row, results are displayed for the IS-divergence, the lower row displays results for the KL-divergence. In each panel, the combined precision CP_i can be compared for different numbers of spectral templates.

to the algorithm described in Section 3.4.1 enables a more accurate modelling and a better discrimination of the instruments.

3.4.2.2 Experiment 2: Real case scenario

In the second experiment, the performance of a semi-automatic transcription system was evaluated in a more realistic scenario. It was assumed that the user had labelled only a subset of the notes for each instrument. These notes were used to estimate template spectra at the corresponding pitches. The template spectra were then used to build complete timbre models for the instruments which were then applied to the remainder of the piece in order to obtain the gain structures.

Experimental setup

For this experiment, each mixture signal in the dataset was split in two halves, each containing approximately 15 s of audio. It was assumed that the user had labelled all notes of all instruments in the *first half* which were used to learn the basis functions as described in Section 3.4.1. The basis functions were

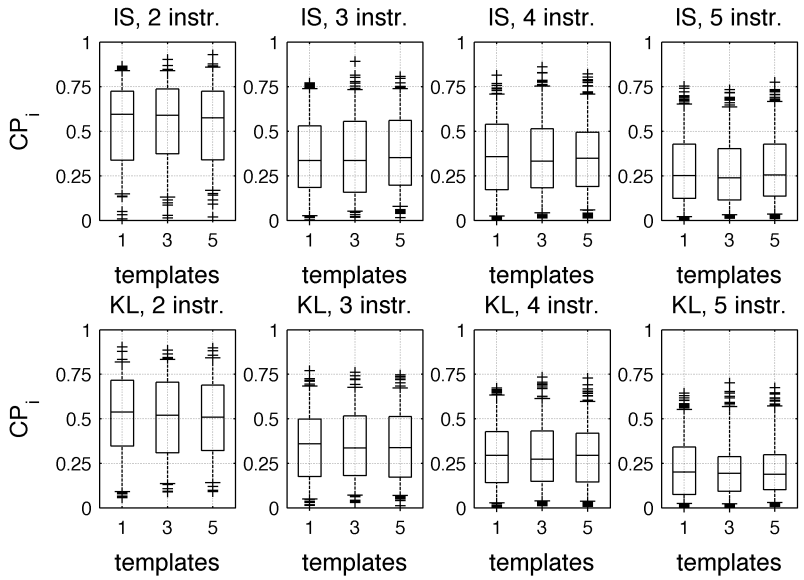


Figure 3.10: Evaluation results for experiment 2. Panels from left to right show the results for the different polyphony levels. In the upper row, results are displayed for the IS-divergence, the lower row displays results for the KL-divergence. In each panel, the combined precision CP_i can be compared for different numbers of spectral templates.

then copied to the surrounding pitches to cover the whole pitch range and were applied to estimate the gains of the *second half*.

As in the first experiment, all combinations of cost functions (IS and KL), numbers of instruments (2–5) and number of templates per pitch (1,3 and 5) were evaluated. Again, 50 iterations of the learning algorithm and 10 iterations for the estimation of the gain matrices were applied.

Results

Figure 3.10 shows the results for the second experiment. The structure of this figure is the same as in Fig. 3.9 in Section 3.4.2.1.

The results of this experiment differ from the results of the previous experiment. In general, there is a larger variance in the results for each configuration. Several trends are clearly visible in the diagram: For both cost functions, the accuracy decreases when the polyphony is increased. The impression from the first experiment that the IS-divergence generally yields better results than the KL-divergence is here confirmed.

In terms of the different numbers of templates per pitch, the results for 1, 3 and 5 templates consistently stay in the same range and no clear trend can be

found. It has to be considered here that the results of this experiment are not only influenced by the number of templates, but also by the fact that templates of those pitches for which no annotation was provided were estimated by replicating the spectra of adjacent pitches. In this experiment, the error introduced by this approximation outweighs the gain of having multiple templates per pitch.

3.5 Summary and discussion

In this chapter, a non-negative analysis framework was presented which is used throughout the thesis. It allows us to incorporate prior information from a human user and to represent this information in a structured way. Furthermore, it enables the estimation and application of timbre models, that is, sets of prototypical spectra for each pitch for all instruments in the recording.

We performed an empirical comparison of two types of user information which enable the use of different timbre models: 1. user information about the instrument identities, which enables the use of timbre models derived from an instrument database, and 2. asking a user to annotate notes for each instrument in the recording, which allows us to extract instrument spectra from the recording itself and to build timbre models that are tailored to the specific instruments in the mixture. The results of this comparison clearly showed that the specific instrument templates outperform the generic timbre models.

Based on these results, the second timbre model was further investigated and a method was proposed that extracts multiple spectral templates for each instrument and pitch. A first experiment confirmed that this method can enhance the quality of the gain matrices if sufficient user information is provided. In a second experiment with limited user information, the results showed that this improvement is outweighed by inaccurate estimations of spectra for which no note annotations were provided.

This second experiment revealed that the quality of the gain matrices can be affected by the estimation of spectral templates for which no note annotations exist. These templates were estimated from templates of other notes. The following chapter therefore looks at different ways of estimating templates where no user-annotation was provided. We will refer to these templates as *missing* spectral templates and we will experimentally evaluate several estimation techniques of these missing templates.

Chapter 4

Missing template estimation

A user-assisted transcription system can only be of practical use if the required amount of user information can be provided by the user in a limited time and with reasonable effort. In the previous chapter, timbre models extracted from a mixture of instruments were shown to obtain more accurate results for the initial pitch analysis than generic timbre models extracted from an instrument database. The extraction of these timbre models from the recording, however, requires user-annotations of notes over a wide pitch range for each individual instrument, which can be a tedious task.

In this chapter we focus on methods that limit the amount of information a user has to provide. In the particular case of extracting timbre models from the recording itself, a reduced set of note annotations from the user will lead to incomplete timbre models, that is, timbre models in which spectral templates cannot be estimated at all relevant pitches due to a lack of annotations at these pitches. Spectral templates at those pitches will be referred to as *missing templates*. Figure 4.1 displays an example of such an incomplete timbre model, in which spectra could only be inferred at a limited number of pitches. In the non-negative analysis framework (Section 3.2), note activations can only be obtained when a corresponding spectral template is available.

In order to obtain complete timbre models, the missing templates have to be estimated based on those templates that could be extracted from the annotations. In the following section (Section 4.1) several methods are discussed that infer missing spectral templates from the provided templates. The methods include purely data-driven techniques as well as some more elaborate instrument models.

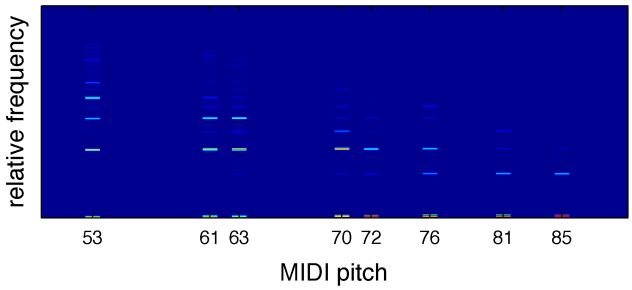


Figure 4.1: Example of an incomplete timbre model in which spectral templates are only present at a limited number of pitches.

In Section 4.2 the estimation methods are experimentally compared and the results are summarised in Section 4.3.

4.1 Estimation methods

In the following sections, several methods for the estimation of missing templates are discussed. In all cases it is assumed that a few typical spectra at different pitches of an instrument are known and that the remaining spectra need to be estimated. In Sect. 4.1.1 and 4.1.2 a few purely data-driven methods are discussed. In Section 4.1.3, the widely used source-filter model is reviewed and an implementation is introduced that does not rely on white excitation spectra. In Section 4.1.4, a method is proposed that adapts previously learned instrument spectra to the known templates.

4.1.1 Copying spectra

A simple method to derive spectral templates at missing pitches is to employ a translated version of the user-provided spectrum at the nearest pitch. Within the basis function structure $\mathbf{W}^{\phi,i}$ of the non-negative analysis framework (cf. Section 3.2) in which spectra are represented by their relative frequencies, this can be achieved by copying the spectra to their adjacent pitch positions. This method assumes that the partial amplitudes of near pitches are approximately the same, which is also the underlying principle of the shift-invariant NMF algorithm in Section 2.4.3.2. An example of a complete timbre model obtained by this method is illustrated in Fig. 4.2. Figure 4.2a shows an incomplete set of spectral templates and Fig. 4.2b shows the spectra obtained by copying.

4.1.2 Interpolating spectra

Another data-driven approach is to estimate the missing spectra by interpolating between the existing spectra. We examined two different interpolation methods.

Plain interpolation

The easiest way to interpolate missing spectral templates is to apply linear interpolation to each spectral bin of the aligned spectra in matrix $\mathbf{W}^{\phi,i}$. The interpolation can thus be applied along the pitch axis to each frequency bin separately. This method is illustrated in Fig. 4.2c for the same incomplete set of spectra in Fig. 4.2a.

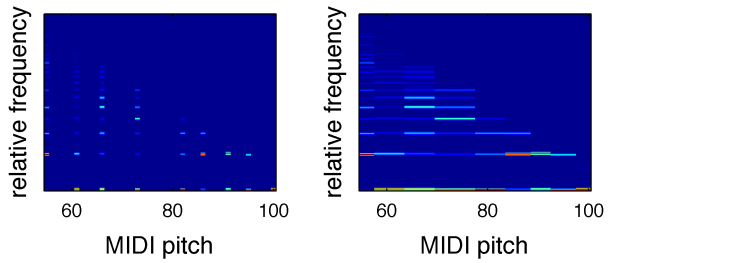
Interpolation with Hann window

Another interpolation method takes several spectra in the pitch vicinity of the missing spectrum into account. For each missing template, a weighted average of surrounding templates is computed using a Hann window centred at the missing template. A window length of 9 semitones was empirically chosen. An example of such a Hann mask is displayed in Fig. 4.2e, where dark areas correspond to larger weights and light areas to lower weights. When no provided spectrum falls within the Hann window range, the missing spectra are estimated by the method of copying spectra (cf. Section 4.1.1). Figure 4.2d shows an example of this method.

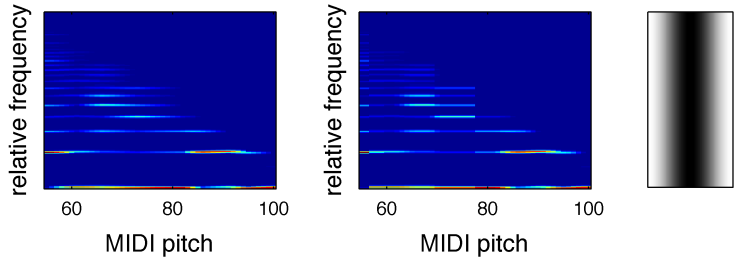
4.1.3 Source-filter model

The source-filter model provides another way of estimating missing spectral templates. It was originally introduced for speech synthesis (Dudley, 1939) but it has also been used extensively for musical instrument modelling both for analysis and synthesis purposes (Virtanen and Klapuri, 2006; Klapuri, 2007; Heittola et al., 2009; Hahn et al., 2010; Hennequin et al., 2011; Caetano and Rodet, 2012). A good overview of existing approaches for source-filter modelling of acoustical instruments was provided by Välimäki et al. (2006).

The source-filter model assumes that the sound production process of an acoustic source consists of two distinct parts: A *generator* or *source* that produces an excitation signal, and a *resonator* or *filter* that shapes the excitation signal. In speech production, the vocal chords are identified as the generator, and the vocal tract is assumed to act as a time-varying filter that modifies the excitation signal. Musical instruments can to some degree be characterised by the same model: the sound production mechanism (e. g. the mouthpiece of a wind instrument or



(a) Incomplete timbre model (b) Copying method



(c) Linear interpolation method (d) Hann-interpolation method (e) Hann mask

Figure 4.2: Illustration of the data-driven missing template estimation methods.

a vibrating string) is modelled by the generator part, and the filter represents the physical resonator (e. g. the instrument body or the sound board).

This unidirectional two-part model does not hold equally well for all types of musical instruments (Välimäki et al., 2006). It is a good fit for bowed string instruments (i. e. the violin family), in which source and filter — the vibrating string and the instrument body — are reasonably well decoupled. It holds less for instruments with stronger interdependencies between the source and the filter part such as many woodwind and brass instruments. These instruments consist of mechanical systems that are strongly coupled, which means that the resonator feeds back and influences the behaviour of the generator. In the case of the clarinet for example, the reed itself would not be able to oscillate at an audible frequency without the support of the resonating tube. Nevertheless, the source-filter model is able to capture instrument characteristics that can be modelled as a function of partial index (e. g. weak even harmonics in clarinet spectra) or absolute frequency (e. g. formants and resonances of the instrument body).

Some computational audio analysis techniques such as *linear prediction* (Atal and Hanauer, 1971), *cepstral representations* (Bogert et al., 1963) or the *True Envelope Estimator* (Röbel, 2010) restrict themselves to extracting the *spectral envelope* from the observed spectra. The concept of a spectral envelope assumes

that the excitation spectrum is white and that the overall shape of the frequency response is purely influenced by the filter model. Röbel (2010) points out that this assumption is an oversimplification, which can be demonstrated by a simple visualisation of spectra of an instrument at different pitches.

The source-filter model has been integrated into the NMF framework in different ways in order to reduce the number of free parameters. Virtanen and Klapuri (2006) replaced the expression for the basis functions in the NMF model (cf. Eq. (2.2)) by the product of an excitation spectrum and a filter spectrum. Heittola et al. (2009) proposed a similar approach in which the filter is modelled by a weighted sum of triangular bandpass filters. Hennequin et al. (2011) incorporate a time varying filter in the NMF framework and model this filter by an autoregressive-moving-average (ARMA) process.

In the experiments in this section, the source-filter model is applied for the estimation of missing instrument spectra. A method is proposed that estimates the model parameters based on the β-divergence between the original and the modelled spectra and operates on isolated instrument spectra as opposed to being integrated in an NMF framework. It enables the estimation of non-white excitation spectra and is inspired by the methods proposed by Virtanen and Klapuri (2006) and Klapuri (2007).

Model formulation

The source-filter model proposed here approximates the excitation spectrum $\mathbf{e}$ and the filter spectrum $\mathbf{h}$ from a number of provided instrument spectra $\mathbf{w}_d$ with $d \in [0, \ldots, D-1]$ at different pitches ϕ_d. $\mathbf{w}_d$ contains spectra on an absolute and logarithmic frequency scale, that is, spectra are here *not* aligned as in the non-negative analysis framework (Section 3.2). The model is described by the following equation:

$$\mathbf{w}_d \approx \hat{\mathbf{w}}_d = s_d \cdot \overset{\phi_d\downarrow}{\mathbf{e}} \bullet \mathbf{h}. \tag{4.1}$$

In this equation, the $\bullet$ operator denotes elementwise multiplication of the vectors. s_d is a scaling factor that compensates for gain differences among the provided instrument spectra. The pitch ϕ_d of each spectrum $\mathbf{w}_d$ is here expressed in terms of frequency bin indices of the fundamental frequency. The operator $\phi_d\downarrow$ translates the excitation spectrum along the logarithmic frequency axis to the correct pitch position ϕ_d. The scaling factors s_d for all pitches can be combined into a single vector $\mathbf{s}$ of length D. This model contains a few ambiguities which need to be addressed in order to provide unique results for $\mathbf{s}$, $\mathbf{e}$ and $\mathbf{h}$. Details and ways to resolve these ambiguities can be found in Appendix B.2.

All vectors $\mathbf{w}_d$ from which $\mathbf{s}$, $\mathbf{e}$ and $\mathbf{h}$ are estimated can be combined into a matrix $\mathbf{W}' \in \mathcal{R}_+^{K,D}$. Likewise, $\hat{\mathbf{W}}'$ denotes a matrix with the same dimensions

that contains in its columns all estimated templates $\hat{\mathbf{w}}_d$ based on the current estimates of $\mathbf{s}$, $\mathbf{e}$ and $\mathbf{h}$ according to Eq. (4.1).

Update equations

Based on the provided instrument spectra at distinct pitches, the model estimates the three vectors $\mathbf{s}$, $\mathbf{e}$ and $\mathbf{h}$. This is achieved by randomly initialising the three vectors and iteratively applying gradient descent on each vector. The β-divergence is used as cost function. For the evaluation in Section 4.2, β was set to 0.

The update equations for the individual components of $\mathbf{s}$, $\mathbf{e}$ and $\mathbf{h}$ are given by

$$s_d \leftarrow s_d \cdot \frac{\sum_{k=0}^{K-1} {W'}_{k,d} {\hat{W}'}_{k,d}^{\beta-2} e_{k-\phi_d} h_k}{\sum_{k=0}^{K-1} {\hat{W}'}_{k,d}^{\beta-1} e_{k-\phi_d} h_k} \tag{4.2}$$

$$e_k \leftarrow e_k \cdot \frac{\sum_{d=0}^{D-1} {W'}_{k+\phi_d,d} \cdot {\hat{W}'}_{k+\phi_d,d}^{\beta-2} \cdot s_d \cdot h_{k+\phi_d}}{\sum_{d=0}^{D-1} {\hat{W}'}_{k+\phi_d,d}^{\beta-1} \cdot s_d \cdot h_{k+\phi_d}} \tag{4.3}$$

$$h_k \leftarrow h_k \cdot \frac{\sum_{\{d|\phi_d \leq k\}} {W'}_{k,d} \cdot {\hat{W}'}_{k,d}^{\beta-2} \cdot s_d \cdot e_{k-\phi_d}}{\sum_{\{d|\phi_d \leq k\}} {\hat{W}'}_{k,d}^{\beta-1} \cdot s_d \cdot e_{k-\phi_d}} \tag{4.4}$$

In these equations, all subscript indices refer to the individual matrix or vector elements. A detailed derivation of these update equations can be found in Appendix B.1.

Postprocessing

The filter response $\mathbf{h}$ can only be reliably estimated at the frequency positions of the harmonic partials of the provided spectra, since not much energy is available between these frequency positions. In a practical setting, only a small number of instrument spectra are provided from which the model parameters need to be inferred. This means that the filter curve can only be reliably estimated at distinct frequency positions. Amplitudes in between those need to be interpolated. Furthermore, depending on the number of partials that fall at a particular frequency bin of $\mathbf{h}$, the filter response might only be estimated based on a few or even a single partial amplitude only, which can result in quite extreme values of the filter response.

One possibility to address both issues is to fit a number of cosine functions to the filter response. This can be done by applying the *discrete cepstrum spectral envelope* to the estimated amplitudes as proposed by Schwarz (1998). This method estimates the amplitudes of the cosine functions based on filter amplitudes at the estimated frequency positions. In order to control the steepness of the fitted curve, the number of cosines can be limited or a regularisation parameter can be integrated (Schwarz, 1998). In our implementation we employed 40 cosine functions and a regularisation parameter of 0.0005, which were empirically found to provide a good approximation of the available data points and a slight smoothing of the filter curve.

To verify the functionality of the model, artificial sets of instrument spectra were created that followed the model assumptions in Eq. 4.1. A harmonic excitation spectrum and a sinusoidal filter curve were selected in order to create spectra at different pitches. The source-filter model was subsequently applied to the generated spectra, the retrieved excitation and filter spectrum were compared to the original ones and the identity was confirmed.

Figure 4.3 displays the results of the estimation of the excitation signal $\mathbf{e}$ and the filter response $\mathbf{h}$ for a violin and a clarinet. The violin excitation spectrum exhibits the typical exponential decay of a sawtooth wave which is caused by the friction of the bow and the restoring force of the string. The filter spectrum contains the typical 'f-hole resonance' at about 270 Hz, a corpus resonance at about 540 Hz, and the typical decay below 250 Hz (Fletcher and Rossing, 1991, p. 247). The excitation spectrum of the clarinet captures the weak even harmonics due to the construction as a closed pipe. An implementation of this source-filter model is available from `http://code.soundsoftware.ac.uk/projects/sourcefiltermodel`.

4.1.4 Adapting database templates

In Chapter 3 it was shown that considerably higher transcription accuracies can be achieved when spectral templates are learned directly from the recording under analysis as opposed to a database of instruments. However, database templates might be useful for the estimation of spectra at missing pitches, as they can provide evidence about typical spectra of the instruments without employing an explicit instrument model. The reason for the different results of database templates and extracted templates in Chapter 3 can be seen in the varying recording conditions and the differences in the construction of the instruments. We assume here that these differences can each be described by a linear time-invariant (LTI) system, and their sequential application can be summarised by a single LTI system. This LTI filter can be estimated by a

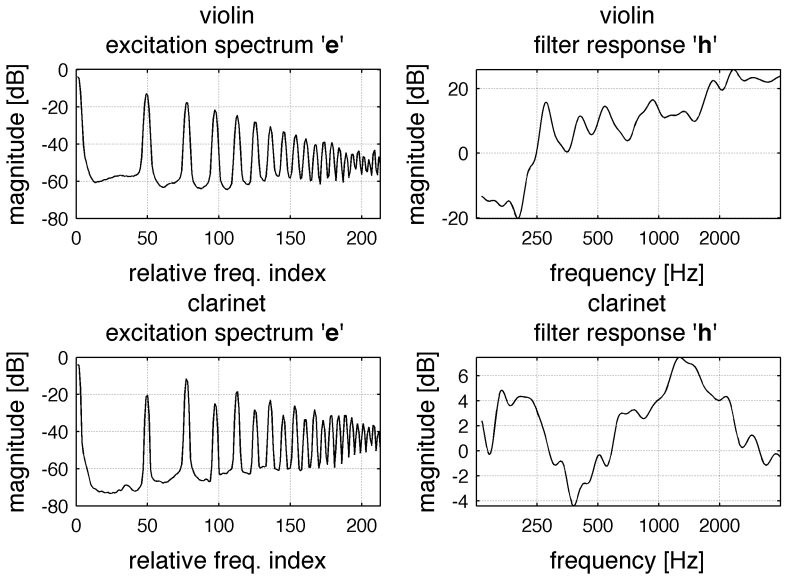


Figure 4.3: Excitation signal **e** and filter response **h** estimated for the spectra of a violin and a clarinet.

comparison of the extracted spectra and the database spectra and it can be used to adapt the database spectra at missing pitches.

The concept of adapting instrument spectra by a static filter has been proposed in previous work. Ozerov et al. (2005) use a GMM-based method to separate a singing voice from the accompaniment. A Maximum Likelihood Linear Regression (MLLR) method is employed to estimate a filter that adapts the generic voice model to the specific voice in the recording. The approach by Jaureguiberry et al. (2011) extends the basic NMF model by diagonal matrices containing the magnitude response of the adaptation filter for each instrument in the recording and present an algorithm for the joint estimation of all instrument filters given the fixed dictionaries of the instruments.

The template adaptation proposed here is similar to the one proposed by Jaureguiberry et al. (2011). Our approach works on isolated instrument spectra rather than being integrated in an NMF model. It estimates the filter frequency by minimising the β-divergence between the already extracted spectra and the filtered database templates, and it applies a smoothing in order to estimate the filter gains at frequency positions where no significant partial energy is present. An implementation of the algorithm is available from: `http://code.soundsoftware.ac.uk/projects/adaptinstrspec`.

Model formulation

In mathematical form, the adaptation of database spectra to the spectra of the recording can be expressed by

$$\mathbf{w}_{\text{data},d} \approx \hat{\mathbf{w}}_{\text{data},d} = \mathbf{w}_{\text{DB},d} \bullet \mathbf{f}. \tag{4.5}$$

In this equation, $\mathbf{w}_{\text{data},d}$ denotes the spectra estimated from the recording at pitches ϕ_d with $d \in [1, \ldots, D]$. $\hat{\mathbf{w}}_{\text{data},d}$ is the approximation of these spectra resulting from the elementwise multiplication of the database spectra $\mathbf{w}_{\text{DB},d}$ with the filter response $\mathbf{f}$. In the same way as in Section 4.1.3, $\mathbf{w}_{\text{data},d}$ contains the spectra on an *absolute* frequency scale as opposed to the relative scale in the non-negative analysis framework (Section 3.2).

Filter estimation

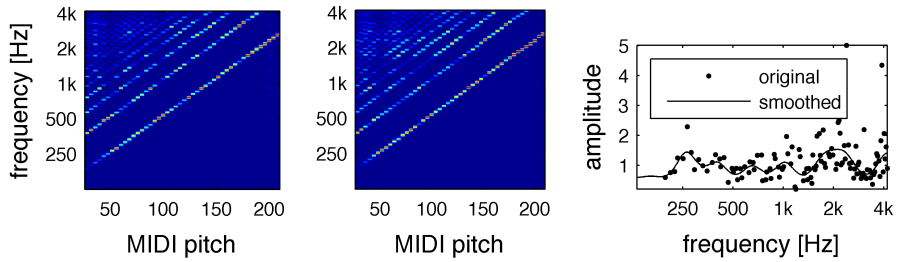
The aim of the estimation procedure is to determine $\mathbf{f}$ in such a way that the error between the original spectra and the filtered database spectra is minimised. The β-divergence (Section 2.4.1) is again applied which generalises various well known cost functions. We use $\beta = 0$ for the evaluation in Section 4.2. The filter response f_n can be estimated by the following equation:

$$f_k = \frac{\sum_{d=0}^{D-1} w_{\text{data},d,k} \hat{w}_{\text{DB},d,k}^{\beta-1}}{\sum_{d=0}^{D-1} \hat{w}_{\text{DB},d,k}^{\beta}} \tag{4.6}$$

The terms $w_{\text{data},d,k}$ and $w_{\text{DB},d,k}$ refer to the k-th element in vectors $\mathbf{w}_{\text{data},d}$ and $\mathbf{w}_{\text{DB},d}$. A derivation of this equation is provided in Appendix C.

Postprocessing

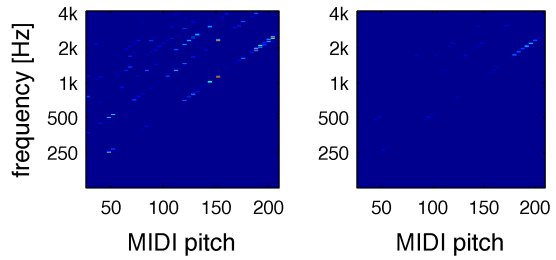
For the estimation of $\mathbf{f}$, only the peak amplitudes of the partials are considered. Here, the same problem arises as in the case of the source-filter model in Section 4.1.3: if the number of spectra $\mathbf{w}_{\text{data},d}$ is small, $\mathbf{f}$ will not be estimated at all frequency bins k and the filter response needs to be interpolated. The same cosine approximation of the filter response as in Section 4.1.3 is therefore applied to interpolate and smooth the filter response. In this case, 20 cosine coefficients are used and a regularisation parameter of 0.001 is applied. These parameters were found to lead to a similar tradeoff between approximation accuracy and smoothing as in the postprocessing step of the source-filter model.



(a) Spectra of violin 1

(b) Spectra of violin 2

(c) Estimated filter $\mathbf{f}$



(d) β-divergence between original spectra

(e) β-divergence between adapted spectra

Figure 4.4: Example adaptation of two different sets of violin spectra.

In the same way as for the source-filter model above, the functionality of the filter estimation was initially verified. Sets of instrument spectra were filtered by a sinusoidal filter response in order to simulate spectra of the same instrument in different recording conditions. The filter curve was then estimated based on a comparison between the original spectra and the generated spectra. In all cases, the estimated filter was identical to the filter that was used to generate the spectra.

An example of an adaptation of two different sets of instrument spectra is illustrated in Fig. 4.4. Figures 4.4a and 4.4b show sets of spectral templates of two different violins. The estimated adaptation filter according to Eq. (4.6) can be seen in Fig. 4.4c. The dots indicate the filter gains that have been estimated at the peaks of the partials, that is, at the points with significant energy. The line shows the filter curve after postprocessing (interpolation and smoothing). In Figs. 4.4d and 4.4e the errors between the original sets of spectra and the adapted sets of spectra can be compared. The color scaling in these two figures is identical, which makes it obvious that the adaptation filter significantly reduces the differences between the two sets of spectra. The total energy of the β-divergences in this example is reduced by a factor of 8.3.

4.2 Evaluation

The estimation methods described in Sections 4.1.1–4.1.4 were experimentally compared. It was investigated how accurately each set of estimated spectra is capable of representing the actual instruments in a transcription context.

4.2.1 Datasets

The datasets employed for this experiment were the MIREX multiple-f0 estimation dataset, the Bach10 dataset and the Trios dataset. All these datasets consist of recordings with multiple instruments and are therefore suitable for the evaluation of transcription algorithms that aim at transcribing the individual instrument parts.

MIREX multi-f0 development set

The MIREX multi-f0 development set[1] was originally released in a shorter version as a development dataset for the *Multiple Fundamental Frequency Estimation & Tracking* task of the Music Information Retrieval Evaluation eXchange (MIREX) and was later extended by Benetos and Grindlay (Benetos and Dixon, 2011b). It consists of a single recording of an excerpt of the 3rd movement of Beethoven's String Quartet Op. 18 No. 5 arranged for a wind quintet. The wind quintet consists of the instruments: flute, oboe, clarinet, french horn and bassoon. The arrangement of the 4 instrument parts for a quintet entails that 2 instruments — the oboe and the clarinet — largely play the same melodic part an octave apart. As indicated in Section 2.2.8, this complicates the transcription task since the partials of the oboe are completely overlapped by the partials of the clarinet. The length of the piece amounts to 54 s.

Bach10 dataset

The Bach10 dataset[2] was released in May 2012 (Duan et al., 2010). It contains recordings of ten 4-part chorales from J. S. Bach played by 4 different instruments: violin, clarinet, alto sax and bassoon. The recordings are between 25 s and 41 s long.

Trios dataset

The Trios dataset[3] was produced by Fritsch (2012). As the name implies, it contains five trio recordings of various instrumentations. The trios are excerpts

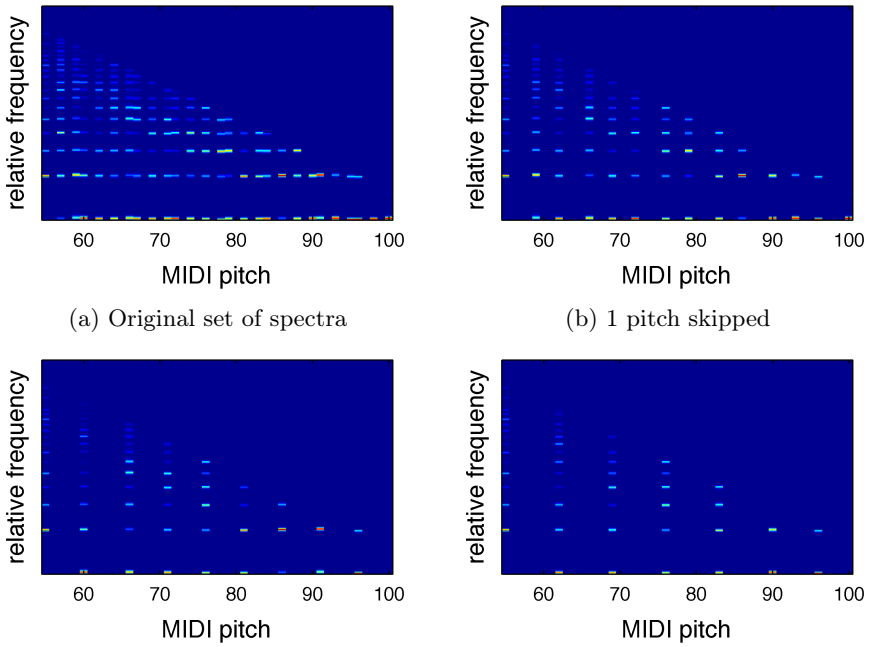
[1] available from: `http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/12`
[2] available from: `http://music.cs.northwestern.edu/`
[3] available from: `http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/27`

of classical music pieces by Mozart, Schubert, Brahms and Lussier, as well as the jazz standard 'Take Five' by Paul Desmond. Each piece has a different instrumentation: clarinet, viola and piano for the Mozart piece, violin, cello and piano for the Schubert piece, violin, french horn and piano for the Brahms trio, trumpet, bassoon and piano for the Lussier piece and finally alto sax, piano and drums for Take Five. For the experiments in this chapter, only the first four recordings were used, because Take Five contains unpitched instrument sounds which are not in the scope of these experiments.

4.2.2 Experimental setup

Based on the ground truth information, spectral templates were extracted based on the information about all notes of all instruments for each recording in the datasets. From these sets of templates, several templates were systematically discarded for each instrument. This process is illustrated in Fig. 4.5: Fig. 4.5a displays a timbre model extracted from a recording based on the ground truth note annotations. Figures 4.5b–d display the reduction process: from all the nominal pitches that appear in the signal, only the templates of every second, third, fourth, etc. nominal pitch are preserved, while the templates in between are omitted, thereby successively decreasing the density of the original data. For the experiment, skips of 0, 3, 6 and 9 nominal pitches were employed. It is debatable whether this *relative* pitch resolution measure is actually meaningful or whether an absolute measure (e. g. major third resolution, quart resolution, etc.) would be more appropriate. Due to the relatively short length of the pieces and the fact that all pieces were based on diatonic scales, the pitches of the individual instruments did not cover the full chromatic scale which makes the application of an absolute pitch resolution difficult.

All estimation methods were then applied to these reduced sets of instrument spectra and the metrics from Section 3.2.3 were computed based on the full set of estimated spectra for each method. For all analyses, a constant-Q spectrogram with a frequency resolution of four bins per semitone and a time resolution of 4.1 ms was used. The lowest frequency bin was chosen as 32 Hz (corresponding to the pitch C1) and the highest frequency bin was set to 4186 Hz (corresponding to C8). This results in a total number 336 frequency bins. Employing a sub-semitone resolution for the analysis implies that even in the case of 0 skipped pitches (i. e. when no spectra are omitted) spectra in between nominal pitches need to be estimated in order to obtain spectra at each fundamental frequency.



(a) Original set of spectra

(b) 1 pitch skipped

(c) 2 pitches skipped

(d) 3 pitches skipped

Figure 4.5: Reduction of a timbre model for the experimental evaluation of the missing template estimation methods.

4.2.3 Results

Figure 4.6 displays the results of the evaluation of the different estimation methods. The results for the different datasets are displayed in each row of panels. Each column contains the results for different numbers of *skipped* pitches — from 0 skipped pitches in the leftmost column to 9 skipped pitches in the rightmost column. Within each panel, each boxplot combines the results for all instruments of all files in the dataset. The *combined precision* CP_i is presented here which combines the gain precision and gain recall measures (cf. Section 3.2.3). The boxplots have the same characteristics as in Section 3.3.3.2 and display the results for the different estimation methods: copying (CPY), interpolation (INT), interpolation with Hann window (HAN), source-filter model (SFM) and adapting database templates (ADT).

In the case where notes are labelled at a high pitch resolution (0 skipped pitches), the purely data-driven methods *copying* (CPY) and *interpolation* (INT) obtain the highest values for CP_i. As indicated in the previous section, even in this case, where none of the provided spectra are discarded, an estimation of missing spectra is required which explains the slight differences in accuracy among these two methods. The remaining estimations methods (HAN, SFM

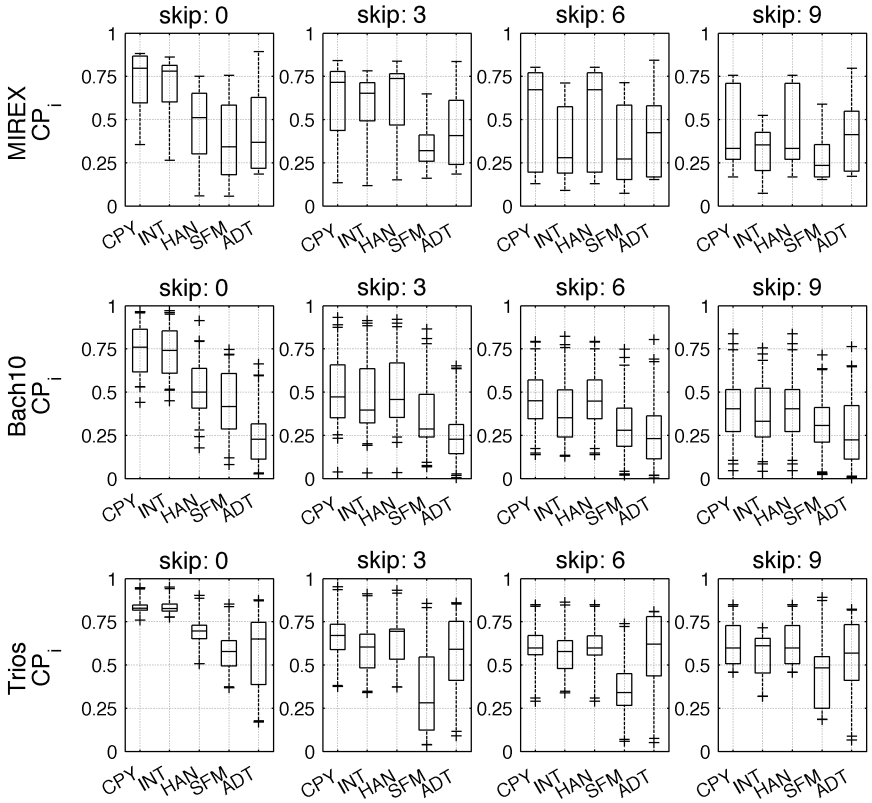


Figure 4.6: Evaluation of different methods for the estimation of missing spectral templates: copying (CPY), interpolation (INT), interpolation with Hann window (HAN), source-filter model (SFM) and adapting database templates (ADT). The results show the combined precision CP_i for different datasets (top to bottom) and different numbers of omitted pitches (left to right).

and ADT) modify even the provided spectra in different ways, which affects the transcription accuracy and leads to worse results: the Hann-interpolation takes several spectra into account and thus also computes a weighted average at the position of the provided pitches. The source-filter model computes new spectral estimates at the given pitches based on the model parameters, and likewise the filtered database spectra replace the original spectra based on the derived filter response.

As the pitch resolution decreases (3, 6 and 9 skipped pitches), transcription accuracies in general decrease, but particularly for the data-driven methods

(CPY, INT and HAN). In some cases, data-driven methods are outperformed by the database template adaptation method (ADT).

In many cases, the *source-filter model* (SFM) surprisingly produces the lowest transcription accuracies. This is particularly the case in the MIREX dataset and in the Trios dataset. In the MIREX dataset, all instruments are wind instruments, for which the source-filter assumption does not hold very well. In the Bach10 dataset, a violin — for which the source-filter model is a better assumption — is present in every quartet file, which might explain the better accuracy. In addition, at low pitch resolutions the small number of available spectra does not allow for an accurate estimation of the source and filter parts.

The results of the method of *adapting database templates* (ADT) in general show the least variation among the different methods when the number of provided spectra varies. Since a reasonable estimate for the spectral shapes is already given by the database templates, the pitch resolution only determines the number of spectra that are available for estimating the filter response $\mathbf{f}$ in Eq. (4.6), which makes this method relatively robust to lower pitch resolutions.

4.3 Summary and discussion

In this chapter, the extraction of timbre models based on user-provided annotations was further investigated. More precisely we looked at ways to minimise the amount of information the user has to provide — particularly the pitch resolution at which notes should be labelled. Different methods for the estimation of template spectra at pitch positions that have not been provided by the user were experimentally compared: data-driven methods, such as copying existing spectra to adjacent pitch positions or interpolating partial amplitudes have been compared to more refined instrument models, such as the source-filter model and an adaptation of pre-learned spectra of the same instrument type. The methods were experimentally compared on three different datasets and with varying pitch resolutions. The results suggest that the data-driven methods *copying* and *interpolation* work well when instrument templates are available at a higher pitch resolution where each template only needs to be used within a comparably small pitch range. At lower pitch resolutions, most of the methods showed significant decreases in performance. The method of adapting previously learned database templates — even though it generally did not exhibit the best results — was least affected by different pitch resolutions.

A surprising result of the evaluation is the fact that the source-filter model in many cases did not provide an accurate representation of the instrument timbre and led to considerably less accurate transcription results. The source-filter model and the related concept of the spectral envelope have been successfully

used for synthesis purposes such as formant-preserving pitch shifting (Moulines and Laroche, 1995). The fact that sounds with a common spectral envelope are perceived as coming from the same source, however, does not necessarily entail that all sounds coming from the same source also exhibit the same spectral envelope. As outlined in Section 4.1.3, other researchers likewise found that this model might not be appropriate for all types of instruments.

Chapter 5

Multiple instrument pitch tracking

The previous two chapters looked at ways to improve the quality of the pitch activation function for each instrument in a multi-instrument recording, and to exploit user input for the extraction of timbre models. These pitch activation functions provide a likelihood measure for each pitch on a per-frame basis. A transcription of individual instrument parts, however, consists of note events that can be described by their start time, duration, pitch and instrument assignment.

In this chapter we propose a method to group the values of a pitch activation function into instrument streams, which is denoted as *pitch tracking*. In the same way as in the previous chapters, this chapter focuses on multiple instrument recordings in which instruments can be distinguished by their timbre and the presented method considers the assignment of pitches to the different instruments in the recording, thereby transcribing the individual parts.

The chapter is structured as follows: The subsequent section reviews prior work on multi-instrument pitch and note tracking. In Section 5.2 the multiple-instrument tracking method is described. The preliminary steps of finding candidate instrument combinations is explained and the details of the Viterbi algorithm for pitch tracking is illustrated. Section 5.3 describes the evaluation procedure including the metrics, and presents experimental results. A summary and conclusion is presented in Section 5.4.

5.1 Prior work on pitch and note tracking for multiple instruments

Computational approaches to music transcription have mainly focussed on the extraction of pitch, note onset and note offset information from a performance, without assigning notes to the underlying instruments (Goto, 2004; Klapuri, 2006; Yeh et al., 2010). This is usually done either by thresholding a pitch activation function (Niedermayer, 2008; Grindlay and Ellis, 2011) sometimes followed by heuristic rules to discard unlikely notes (Bello et al., 2006; Dessein et al., 2010), or by more advanced techniques such as HMMs (Poliner and Ellis, 2007a; Ryynänen and Klapuri, 2005) or temporally constrained PLCA models (Benetos and Dixon, 2013).

Only few approaches have addressed the task of additionally assigning the notes to their sound sources (instruments) in order to obtain a parts-based transcription. The transcription of individual instrument parts, however, is crucial for many of the applications mentioned in Section 1.1. In an early paper, Kashino et al. (1995) incorporated a feature-based timbre model in their hypothesis-driven auditory scene analysis system in an attempt to assign detected notes to instruments. Vincent and Rodet (2004) combined independent subspace analysis (ISA) with 2-state HMMs. Instrument spectra were learned from solo recordings and the method was applied to duet recordings. The harmonic-temporal clustering (HTC) algorithm by Kameoka et al. (2007) incorporates explicit parameters for the amplitudes of harmonic partials of each sound source and thus enables an instrument-specific transcription. However, no explicit instrument priors were used in the evaluation and the method was only tested on single-instrument polyphonic material. The approach by Leveau et al. (2008) uses instrument-specific spectra (atoms) in combination with sparse coding to identify notes played by different instruments. Notes are tracked over time in forward and backward direction starting from a seed atom. Duan et al. (2009) proposed a tracking method that clusters frame-based pitch estimates into instrument streams. Similar pitches in consecutive frames are linked to form *notelets* which are subsequently clustered based on their harmonic structure. Grindlay and Ellis (2011) used their eigeninstruments method as a more generalised way of representing instruments to obtain parts-based transcriptions. The instrument-specific activation functions were simply thresholded to obtain note objects for each individual instrument. The standard NMF framework with instrument-specific basis functions is capable of extracting parts-based pitch activations. However, it only relies on spectral similarity and does not involve pitch tracking or other explicit modelling of temporal continuity. Bay et al. (2012) therefore

combined a PLCA model and a subsequent HMM to track individual instruments over time.

It is difficult to assess the quality of each of these multi-instrument note tracking methods for several reasons:

1. The methods were tested on different datasets with varying numbers of instruments. In some cases only qualitative results were presented.

2. The actual pitch or note tracking algorithm is often integrated in a larger transcription framework which makes it impossible to directly compare this particular part of the transcription algorithm for the different methods.

3. In all cases the transcription system is evaluated as a whole without individual results for the note tracking part and other sub-tasks.

The method described in the subsequent sections follows a similar approach as the one proposed by Bay et al. (2012). The Viterbi algorithm is employed to find the most likely path through a number of *candidate instrument combinations* at each time frame. However, a more refined method for computing the transition probabilities between the states of consecutive time frames is used here.

5.2 Pitch tracking framework

The starting point for the proposed pitch tracking method is a pitch activation function as it was used in Chapters 3 and 4. The non-negative analysis framework introduced in Section 3.2 contains individual pitch activation functions for the instruments in the recording which would enable a detection of pitch tracks from these instrument-specific activation functions. The results of the experiments in Chapters 3 and 4, however, showed that activations are sometimes assigned to the wrong instrument due to inaccurate timbre models and/or insufficient expressivity of the analysis framework. This introduces irreversible errors at an early stage which would be propagated to the pitch tracking algorithm.

The multiple instrument pitch tracking method presented in this chapter therefore bases the decision about note parameters not only on the *quality of the reconstruction* of the observed spectra as measured by the cost function of the non-negative analysis framework (cf. Section 2.4.1). It takes into account two additional criteria: the *temporal continuity of the pitches* and explicit *hypotheses about the activity status* of each individual instrument. These criteria are formulated as a combined state transition probability in a Viterbi framework (Section 2.5) which finds the globally optimal decision for a number of candidate assignments of pitches to instruments.

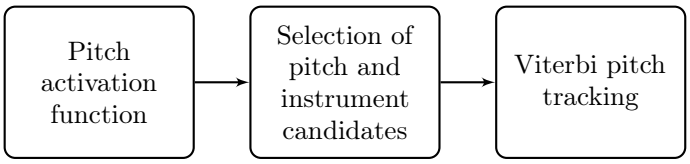


Figure 5.1: Processing stages of the multi-instrument pitch tracking method.

Figure 5.1 illustrates the processing stages of the tracking algorithm. In Sections 5.2.1–5.2.3 each of these stages is described in detail. The method starts by extracting a pitch activation function *without* instrument assignments and detects the most prominent pitches in each time frame. For these pitch candidates several *candidate assignments* to the underlying instruments are selected. Based on this selection, the *Viterbi algorithm* finds the most likely sequence of pitch-instrument assignments based on the criteria mentioned above.

5.2.1 Pitch activation function

In Chapter 3, pitch activations were extracted based on two different types of timbre models. Using timbre models extracted from the recording under analysis were shown to result in higher quality pitch activation functions than generic timbre models, at the cost of requiring a large amount of user input. For the pitch tracking stage, we revert to the generic timbre models which only require knowledge about the identities of the instruments in the recording. This eliminates the problem of missing template estimation and allows us to focus on the evaluation of the proposed pitch tracking method. The generic templates can be assumed to be able to extract the general pitch content reasonably well. For the final decision about assignments, more criteria than just the reconstruction error are considered which makes the choice of this timbre model less critical.

The non-negative analysis framework is used for the initial pitch analysis. The generic basis functions for the instrument in the recording were extracted from the RWC musical instrument database (Goto et al., 2002) in the same way as described in Section 3.3.1. A single template ($R = 1$) per instrument and pitch was extracted for each instrument on a logarithmic frequency scale and with a frequency resolution of four bins per semitone. 20 iterations of the update equation for the pitch activations (Eq. (3.3)) were employed. The instrument-specific pitch activation functions were collapsed to a single activation function $\mathbf{G}$ by summing the activations of all instruments at the same time and pitch index:

$$[\mathbf{G}]_{\phi,n} = \sum_{i=0}^{I-1} \left[H^{\phi,i}\right]_{r,n}, \tag{5.1}$$

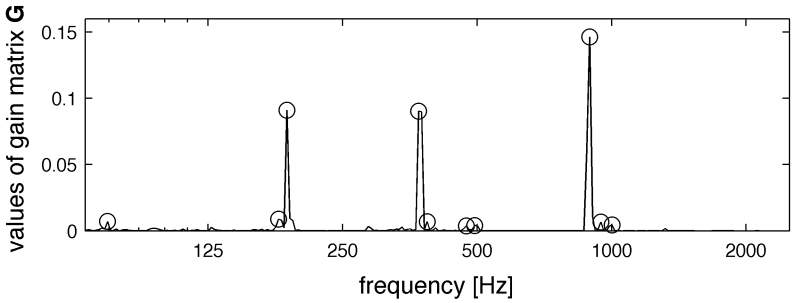


Figure 5.2: Example for picking the M highest peaks ($M = 10$) in a single time frame of the gain matrix $\mathbf{G}$. In this 3-instrument mixture, the bassoon plays a note at pitch F♯3 (188 Hz), the clarinet at F♯4 (375 Hz) and the flute at A5 (893 Hz).

with $r = 1$. It should be pointed out here that the tracking method described below is not specific to the pitch activation function employed here. Numerous other ways of computing pitch activation functions have been proposed (e. g. Klapuri, 2005, 2006; Yeh et al., 2010; Dressler, 2011) which might equally well be used for the initial pitch analysis.

5.2.2 Selection of candidate pitch-instrument combinations

From the pitch activation function, in each time frame the M highest peaks were extracted as candidate pitches. This is illustrated for a single frame of a pitch activation matrix $\mathbf{G}$ in Fig. 5.2. All assignments of peaks to instruments were considered. To make this combinatorial problem tractable it was assumed that each instrument is monophonic and that no two instruments will play the same pitch at the same time. An extension to polyphonic instruments is discussed in Section 5.4. The total number of pitch-to-instrument assignments is given by the following equation:

$$C(M, I) = \frac{M!}{(M - I)!}, \tag{5.2}$$

where M denotes the number of extracted peaks per frame and I the number of instruments. Depending on both M and I, this can lead to a large number of combinations. In practice, however, all combinations for which a peak lies outside the playing range of one or more instruments can be discarded. In our experiments this reduced the overall number of combinations considerably. If *all* peaks lie outside the range of an instrument, however, the case in which the instrument is inactive has to be included. The activity status of an instrument and how it is used in the Viterbi algorithm will be detailed in Section 5.2.3.

In order to reduce the number of combinations for the subsequent Viterbi framework, the reconstruction error was computed for each pitch-instrument combination and only the N_C combinations with the lowest reconstruction error were considered as candidates in the Viterbi framework. To compute the reconstruction error, NMF with fixed instrument spectra was applied in which the basis functions corresponded to the spectra of the instruments at the assigned pitches. The instrument spectra were taken from the generic timbre models (Section 5.2.1). Despite the potentially large number of pitch-instrument combinations, these reconstruction errors can be computed in a reasonable amount of time, since the number of components is very small (less than or equal to I). In the experiments in Section 5.3, just 5 iterations of the NMF update rules (Eq. (2.12)) was sufficient to include the correct pitch-instrument combination in the selection in each time frame. The gains g obtained from these NMF analyses were used for the activity modelling as described in the following section.

5.2.3 Viterbi algorithm

The Viterbi algorithm was employed to find the most likely sequence of pitch-instrument combinations over time. A general introduction to the Viterbi algorithm can be found in Section 2.5.

States

In our framework, a state x at time frame n can mathematically be described as $S_{x,n} = (\phi_{x,n,i}, a_{x,n,i})$ with $i \in \{0, \ldots, I-1\}$. In this formulation, $\phi_{x,n,i}$ denotes the pitch of instrument i and $a_{x,n,i}$ is a binary activity flag that indicates whether the instrument is active at that time frame. The observed gain value for instrument i of a state $S_{x,n}$ is denoted by $g_{x,n,i}$ and the reconstruction error of the state is given by $e_{x,n}$.

The states of the Viterbi algorithm were obtained by considering all hypotheses of instruments being active ($a_{x,n,i} = a$) and inactive ($a_{x,n,i} = \overline{a}$) for each of the selected pitch-instrument combinations from Section 5.2.2. Note that in this process, a large number of duplicates are produced when the pitches of *all active* instruments agree between the selected pitch-instrument combinations. As an example, consider a two-instrument mixture with the following two candidate pitch-instrument combinations at time n: $(\phi_{1,n,1} = x, \phi_{1,n,2} = y)$ and $(\phi_{2,n,1} = x, \phi_{2,n,2} = z)$. The activity hypothesis in which $a_{1,n,1} = a_{2,n,1} = a$ and $a_{1,n,2} = a_{2,n,2} = \overline{a}$ produce Viterbi states that both assume that instrument 1 is responsible for pitch x and that instrument 2 is inactive. In this case, only the state with the lowest reconstruction error $e_{x,n}$ was considered.

Transition probability

For the transition probability from state $S_{y,n-1}$ at the previous frame to state $S_{x,n}$ at the current frame, 3 different criteria were considered:

1. **Probability based on the reconstruction error $\mathbf{p_e(e)}$:** States with lower reconstruction errors $e_{x,n}$ should be favoured over those with higher reconstruction errors. The reconstruction error was therefore modelled by a one-sided normal distribution with zero mean: $p_e(e) = \mathcal{N}(0, \sigma_e^2)$. A plot of this distribution can be found in Fig. 5.3a.

2. **Probability of pitch continuity $\mathbf{p_d}(\phi_\mathbf{n}|\phi_{\mathbf{n-1}})$:** A pitch continuity criterion was applied as proposed by Bay et al. (2012):

$$p_d(\phi_{x,n,i}|\phi_{y,n-1,i}) = \frac{1}{\sigma_d\sqrt{2\pi}} e^{\frac{(\phi_{x,n,i}-\phi_{y,n-1,i})^2}{2\sigma_d^2}}. \tag{5.3}$$

   Large jumps in pitch are thereby discouraged while continuous pitch values in the same range in successive frames are favoured. This criterion accounts for both the within-note continuity as well as the continuity of the melodic phrase. Figure 5.3b illustrates this distribution. This probability was only computed for those instruments that were active in both frames n and $n-1$.

3. **Probability of instrument activity $\mathbf{p_a}(\mathbf{a_n}|\mathbf{g_n}, \mathbf{a_{n-1}})$:** An explicit activity model was employed that expresses the probability of an instrument being active at frame n given its gain at frame n and its activity at the previous frame $n-1$. With Bayes rule, this probability can be expressed as

$$p_a(a_{x,n,i}|g_{x,n,i}, a_{y,n-1,i}) = \frac{p(g_{x,n,i}|a_{x,n,i}, a_{y,n-1,i}) \cdot p(a_{x,n,i}|a_{y,n-1,i})}{p(g_{x,n,i}|a_{y,n-1,i})}. \tag{5.4}$$

   It can be assumed that the gain of an instrument depends only on its activity status at the same time frame and is independent of the activity status of the previous time frame, which leads to the following simplifications:

$$p(g_{x,n,i}|a_{x,n,i}, a_{y,n-1,i}) = p(g_{x,n,i}|a_{x,n,i}) \tag{5.5}$$

$$p(g_{x,n,i}|a_{y,n-1,i}) = p(g_{x,n,i}). \tag{5.6}$$

   No prior assumptions about the values $g_{x,n,i}$ are made and hence $p(g_{x,n,i})$ was modelled as a uniform distribution. This however means that it affects all state transitions in all time frames in the same way and can therefore

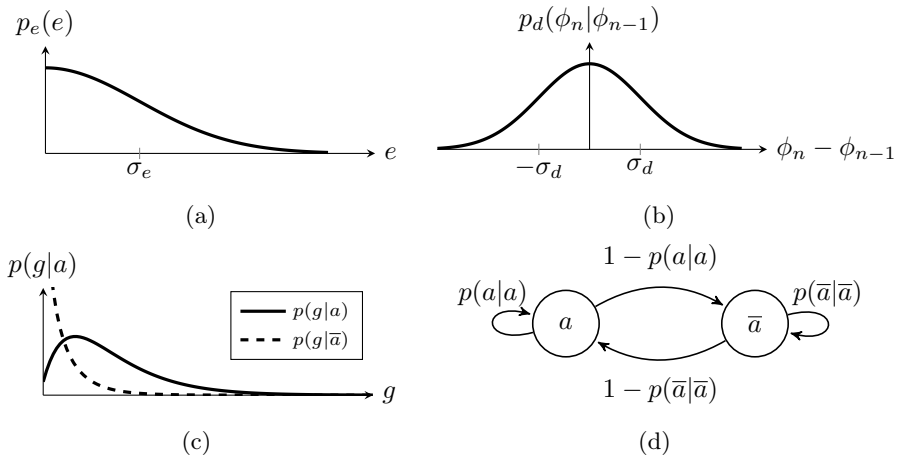


Figure 5.3: Components of the transition probability for the Viterbi algorithm.

be omitted. These assumptions allow us to express Eq. (5.4) in a simpler form as

$$p_a(a_{x,n,i}|g_{x,n,i}, a_{y,n-1,i}) = p(g_{x,n,i}|a_{x,n,i}) \cdot p(a_{x,n,i}|a_{y,n-1,i}). \tag{5.7}$$

The probability $p(g_{x,n,i}|a_{x,n,i})$ was here modelled by two Gamma distributions as illustrated in Fig. 5.3c. The probability $p(a_{x,n,i}|a_{y,n-1,i})$ for transitions between active and inactive states is illustrated in Fig. 5.3d.

Based on these criteria the overall transition probability from state $S_{y,n-1}$ at time $n-1$ to state $S_{x,n}$ at time n can be formulated as the combination of the above probabilities:

$$p(S_{x,n}|S_{y,n-1}) = p_e(e_{x,n}) \cdot \left(\prod_{\substack{\{i|a_{x,n,i}= \\ a_{y,n-1,i}=a\}}} p_d(\phi_{x,n,i}|\phi_{y,n-1,i}) \right) \cdot \left(\prod_{i=1}^{I} p(g_{x,n,i}|a_{x,n,i}) \cdot p(a_{x,n,i}|a_{y,n-1,i}) \right) \tag{5.8}$$

For reasons of numerical accuracy, we computed the log-probability, which is given by

$$\ln[p(S_{x,n}|S_{y,n-1})] = \ln[p_e(e_{x,n})] + \left(\sum_{\substack{\{i|a_{x,n,i}= \\ a_{y,n-1,i}=a\}}} \ln[p_d(\phi_{x,n,i}|\phi_{y,n-1,i})] \right) + \left(\sum_{i=1}^{I} \ln[p(g_{x,n,i}|a_{x,n,i})] + \ln[p(a_{x,n,i}|a_{y,n-1,i})] \right). \quad (5.9)$$

5.3 Evaluation

5.3.1 Metrics

The individual monophonic instrument transcriptions obtained by the multi-instrument pitch tracking procedure can be interpreted as melodies played by that instrument. This enables the application of the metrics for the *MIREX Audio Melody Extraction*[1] task which were likewise used by Grindlay and Ellis (2011) and Bay et al. (2012). These metrics are frame-based measures and contain two different aspects: an evaluation of the activity detection, and an evaluation of the detected pitch at those time frames in which the instrument was correctly detected as active. In the description of the evaluation metrics, activity is denoted as *voicing* and active frames are called *voiced* frames.

The activity component compares the activity labels of the ground truth to those of the algorithmic results. Frames that are labelled as *active* or *inactive* in both the ground truth and the estimate are denoted as *true positives* (TP) and *true negatives* (TN), respectively. If labels differ between ground truth and estimate, they are denoted as *false positives* (FP) or *false negatives* (FN).

The pitch detection component only looks at the *true positives* and measures how many of the pitches were correctly detected. Correctly detected pitches are denoted by TPC, incorrect pitches by TPI, with $TP = TPC + TPI$. A pitch is denoted as correct when its nominal pitch matches the nominal pitch of the ground truth.

[1] `http://www.music-ir.org/mirex/wiki/2012:Audio_Melody_Extraction#Evaluation_Procedures`

From these integer numbers, precision, recall and f-measure are computed in the following ways:

$$\text{precision} = \frac{\sum_{i=0}^{I-1}\sum_{n=0}^{N-1} TPC_{i,n}}{\sum_{i=0}^{I-1}\sum_{n=0}^{N-1} TP_{i,n} + FP_{i,n}} \tag{5.10}$$

$$\text{recall} = \frac{\sum_{i=0}^{I-1}\sum_{n=0}^{N-1} TPC_{i,n}}{\sum_{i=0}^{I-1}\sum_{n=0}^{N-1} TP_{i,n} + FN_{i,n}} \tag{5.11}$$

$$\text{f-measure} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}, \tag{5.12}$$

The precision metric indicates what percentage of the frames that were detected as active had the correct pitch label. This measure reaches its maximum when the number of false positives (FP) is minimised, and when the pitch is correctly detected in all time frames that were detected as active. The recall metric computes the number of correctly detected pitches in relation to the overall number of correct pitches in the ground truth. This measure is maximised when the number of missed active frames is minimised and again when all pitches have the correct pitch label.

Instrument confusion, i. e. the assignment of a pitch to the wrong instrument at a time frame, affects three different quantities: it increases FN of the correct instrument as well as FP of the incorrect instrument, and it decreases TPC of the incorrect instrument. Hence it has an influence on the recall metric of the correct instrument and on both the precision and recall of the incorrect instrument.

5.3.2 Experimental setup

The multi-instrument note tracking algorithm described above was evaluated on the development dataset for the *MIREX Multiple fundamental frequency & estimation task* which was previously described in Section 4.2.1. For this evaluation, all mixtures of 2–5 instruments were created from the separate instrument tracks, which resulted in 10 mixtures of 2 and 3 instruments, 5 mixtures with 4 instruments and a single mixture containing all 5 instruments. A MIDI file associated with each individual instrument provided the ground truth for the pitch tracks.

In the pitch-instrument candidate selection stage (Section 5.2.2), the number of pitch candidates per frame M was set to 10 and 100 candidate pitch-instrument combinations were selected. In our experiments these parameter values included

the correct pitch-instrument combination in the selection in all frames. The value of σ_e was empirically set to a value of 10^{-3}, and likewise σ_d was empirically chosen to be 10 semitones. The parameters for the activity model were extracted from the analysis of the test data set: for $p(g|a)$, the shape and scale parameters of the gamma distribution were set to (2.02, 0.08) and for $p(g|\overline{a})$ to (0.52, 0.07), which resulted in the distributions that can be seen in Fig. 5.3c. The activity transition probabilities $p(a|a)$ and $p(\overline{a}|\overline{a})$ were set to 0.99 and 0.98, respectively, at the given hop size of 4.1 ms.

The note tracking procedure was applied to all instrument mixtures and the evaluation metrics from 5.3.1 were computed for each file individually. In order to gain insights into the contribution of each of the three different parts of the transition probability, the results were first computed using only the reconstruction error criterion p_e as the transition probability, then adding the activity criterion p_a and finally using all three criteria including p_d, i. e. the full transition probability as given in Eq. (5.9).

5.3.3 Results

The results were computed for each file in the test set individually and are shown in Fig. 5.4. Precision, recall and f-measure metrics are reported in the different rows. In each column the results for a different polyphony level are reported. Within each panel, the contributions of the different parts of the transition probability can be compared, starting with only the reconstruction error criterion p_e and successively adding the criteria p_a and p_d. The boxplots have the same characteristics as in Section 3.3.3.2.

The results show a consistent improvement of the f-measure over all polyphony levels when the different parts of the transition probability are successively added. When using only the reconstruction error p_e, all frames are considered as active. Hence, the addition of the activity criterion p_a reduces the number of false positives (FP) and leads to a considerable increase in the precision measure. At the same time, however, it also introduces false negatives (FN) while leaving the pitch estimates (TPC) unchanged which actually results in a slight decrease of the recall measure. The gain in precision, however, outweighs the loss in recall, which results in an overall increase of the combined f-measure. The application of the pitch continuity criterion p_d only affects the rate of correct pitch detections (TPC). The improvement has therefore an effect on both the precision and recall measure. Overall it leads to an increase at the same order of magnitude as the previous addition of the activity criterion. The median f-measure reaches to 0.78 for the 2-instrument mixtures, to 0.58 for mixtures

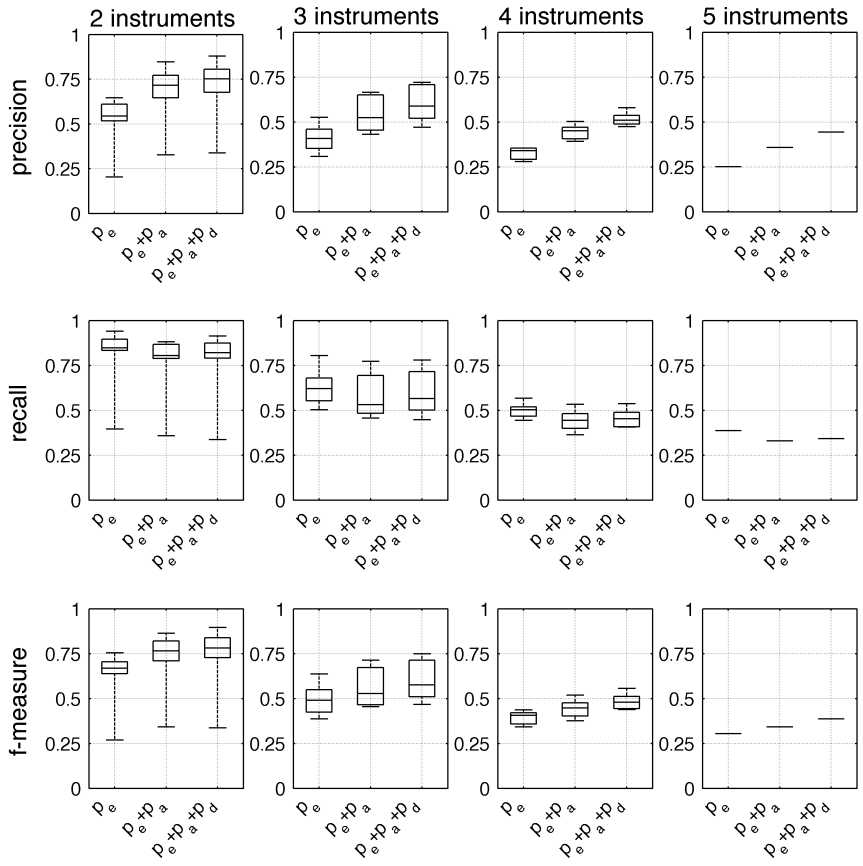


Figure 5.4: Experimental results of the Viterbi note tracking method for different combinations of the transition probability components.

of 3 instruments and up to 0.48 and 0.39 for 4 and 5 instrument mixtures, respectively.

An example result of the note-tracking procedure for a 3-instrument mixture is displayed in Fig. 5.5. In this diagram, the note tracking results are superimposed with the ground truth pitches which enables a more detailed analysis of the types of errors produced by the algorithm. The algorithm misses several notes of the bassoon as well as a few notes of the flute trill. It particularly misses *notes with short durations* that appear in isolation. This has to be attributed to the pitch activation part: if the gain of the notes is comparably weak and the duration is relatively short, an activation of the note is discouraged by this criterion. A second type of error are *instrument confusions* which can be observed in the bassoon part around 4 s. The bassoon part estimate explains the reverberation of some clarinet notes which slightly overlap in time with the subsequent note of

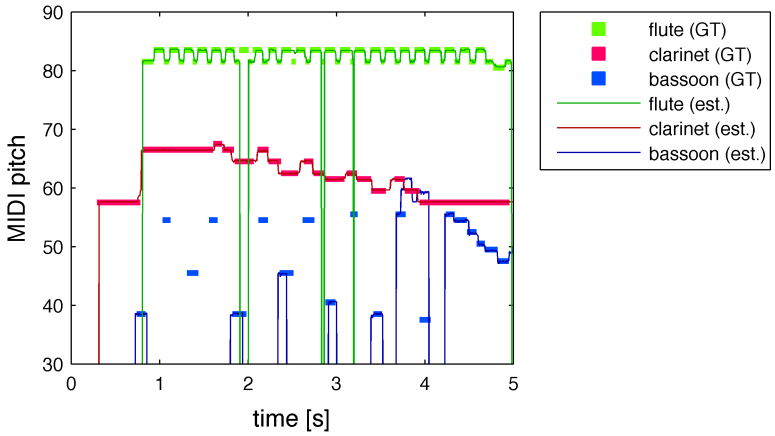


Figure 5.5: Example results of the note tracking method for a mixture of three instruments: bassoon, clarinet and flute. The colored boxes represent the ground truth (GT) note annotations, the lines shows the estimated (est.) note trajectories for each instrument. Inactive instruments are shown as pitch 0.

the clarinet. The inactivity of the bassoon at the beginning of the first confusion allows the algorithm to use its spectra to reduce the reconstruction error at these time frames. The fact that the bassoon was previously active further benefits this error. A third type of error can be found in the *inaccurate estimates of the onset and offset times* of some notes. These errors could be avoided by adjusting the parameters of the instrument activity criterion. It should be mentioned here that these mistakes are not necessarily only caused by the note tracking method, but can also be introduced by inaccuracies in the ground truth annotations which are usually hand-corrected based on visualisations of the frequency content.

In terms of the absolute performance of the tracking method, the results were compared to the results reported by Grindlay and Ellis (2011) and Bay et al. (2012). These authors likewise apply their algorithms to recordings of the same wind quintet piece. The results by Grindlay and Ellis (2011) were computed on the same movement of the quintet. However, ground truth data was only available for the first 22 seconds of the recording at the time the paper was written and the ground truth was only extended more recently. Bay et al. (2012) reported their results on other excerpts from the wind quintet piece that are not publicly available, and five 30 s excerpts were used in the evaluation. Both algorithms use the same metrics as the ones described above, and report the *mean* of the results for the different instrument mixtures. To enable a comparison, we likewise compute the mean values of our results. A comparison

	2 instr.	3 instr.	4 instr.	5 instr.
Grindlay and Ellis (2011)	0.63	0.50	0.43	0.33
Bay et al. (2012)	0.67	**0.60**	0.46	0.38
Viterbi tracking	**0.72**	**0.60**	**0.48**	**0.39**

Table 5.1: Comparison of the average f-measure with other multi-instrument tracking methods on similar datasets.

of the results can be found in Table 5.1. Note that these *mean* values differ slightly from the *median* values in the boxplots in Fig. 5.4.

The comparison shows that the proposed algorithm outperforms the previous methods at almost all polyphony levels. While the results are only slightly better than the results reported by Bay et al. (2012), the difference compared to the method proposed by Grindlay and Ellis (2011) is considerably larger. Grindlay and Ellis (2011) used a simple thresholding on the pitch activations and no temporal dependencies between pitch activations were taken into account which underlines the fact that both an explicit activity model as well as a pitch continuity criterion are useful improvements for instrument tracking methods.

5.4 Summary and discussion

In this chapter, an algorithm was proposed that tracks the individual voices of a multiple instrument mixture over time. After computing a pitch activation function, the algorithm identified the most prominent pitches in each time frame and considered assignments of these pitches to the instruments in the mixture. The reconstruction error was computed for all candidate pitch-instrument combinations. Those combinations with the lowest reconstruction errors were combined with instrument activity hypotheses to form the states of a Viterbi framework in order to find the most likely sequence of pitch-instrument combinations over time. The transition probabilities for the Viterbi algorithm were defined based on three different criteria: the reconstruction error, pitch continuity across frames and an explicit model of active and inactive instruments.

The evaluation results showed that the algorithm outperforms other multi-instrument tracking methods which indicates that the activity model as well as the pitch continuity objective are useful improvements over systems which are based solely on the reconstruction error of the combined spectra.

Although in this chapter the instruments were restricted to be monophonic, the method could be extended to incorporate polyphonic instruments. In this case a maximum number of simultaneous notes N_i would have to be specified for each polyphonic instrument. Instead of assigning each peak of the pitch activation function to a *single* instrument, we would allow up to N_i peaks of the

pitch activation function to be assigned to the polyphonic instrument. If the number of simultaneously played notes of the polyphonic instrument remains constant over time, the Viterbi algorithm would combine the notes closest in pitch into individual note streams associated with the polyphonic instrument. If the polyphony increases, one or more of the inactive note streams would transition from an inactive state to an active state. In the same way, if the polyphony decreases, one or more of the active streams would transition to an inactive state.

A potential improvement could address the complexity of the method, that is, reducing the number of peak-to-instrument assignments which leads to a high computational cost for larger polyphonies. Instead of allowing each peak to be assigned to each instrument, peaks could be assigned to a subset of instruments only, based on the highest per-instrument pitch activations in the initial NMF analysis.

Chapter 6

Instrument models including phase information

The timbre models introduced in Chapter 3 aim at capturing the average magnitude spectra of the different pitches of an instrument. These models were motivated by the fact that the timbre of a pitched musical instrument is to a large part influenced by the amplitude relations of harmonic partials (Section 3.1). Another assumption that is generally made when using spectrogram factorisation techniques such as NMF and PLCA is that the observed magnitude spectra of sound mixtures can be approximated by a superposition of the magnitude spectra of the individual sound sources. Although this assumption provides reasonable analysis results in practice, the linearity only holds for the *complex* coefficients of the STFT. From a signal analysis point of view, instrument models have to consider phase information if the linearity property of the time-frequency analysis is to be satisfied. In this chapter, instrument models are proposed that not only capture the magnitude profiles but also a distinct phase profile for each pitch of an instrument. The term *timbre models* is not appropriate for these models, since the phase information does not have an effect on the perceived timbre. These models are here simply denoted as *instrument models*. User information is again required for the estimation of the magnitude and phase profiles.

6.1 Motivation

Phase information is often discarded in the analysis of harmonic sounds, not only because the human auditory system is considered insensitive to absolute phase shifts of harmonic partials as indicated in Section 3.1, but also because the

magnitude spectrogram is often considered more intuitive and easier to model. For all applications in which sounds have to be synthesised from a time-frequency representation, however, the correct estimation of phase values is crucial in order to avoid artefacts due to phase-incoherent overlap of consecutive time frames. For the task of instrument separation, for example, the phase information for the synthesis of each sound source either has to be estimated from the magnitude spectrogram (Griffin and Lim, 1983), or the phases of the original mixture have to be employed for each source (Virtanen, 2007). Using the mixture phases can lead to reasonable results when the number of sources is small and when most time-frequency bins are mainly influenced by a single source. For higher numbers of sources and larger time-frequency overlap, however, it can lead to cross-talk artefacts.

It is not possible to integrate the instantaneous phase of the harmonic partials into a matrix factorisation framework in the same way as the magnitudes since the phase of the partials is not constant over time. Nevertheless, several approaches have been proposed that consider phase information in NMF models. Parry and Essa (2007) propose a phase-aware non-negative matrix factorisation. The authors model the STFT bins as complex random variables, and assume the phase to be uniformly distributed. Iterative update rules are derived based on this assumption. The update rules, however, still estimate the matrices based on the magnitude spectrogram only. In a similar way, Févotte et al. (2009) show that Itakura-Saito NMF is equivalent to a maximum-likelihood parameter estimation of a sum of complex Gaussian components. The Gaussian components have zero mean and a diagonal covariance matrix, and hence assume a uniformly distributed phase. An attempt to explicitly estimate the phase values of the individual sources was made by Kameoka et al. (2009). Their complex NMF algorithm combines the outer product of each NMF basis function $\mathbf{w}_r$ and gain vector $\mathbf{h}_r^\top$ with a phase spectrogram with the same dimensions as the original spectrogram. Complex NMF is not a complex matrix factorisation technique, but a combination of NMF with time-frequency phase estimates. The algorithm is heavily overparameterised and it can be shown that an initialisation with the mixture phases leaves the phase parameters unaltered (up to $\pm\pi$). Lastly, a high resolution NMF framework has been introduced by Badeau (2011, 2012), in order to properly model both the magnitude and phase of complex or real-valued time-frequency representations. This framework, however, does not take the phase relations of harmonic partials into account.

In this chapter, the relative phase offsets between partials in the sustained part of the sounds of harmonic instruments are exploited as a step towards *complex matrix decomposition*. The concept will be reviewed and illustrated in Section 6.2, where a mathematical formulation is presented. In Section 6.3

the model for a complex matrix decomposition is derived and the parameter estimation equations for the *monophonic* case are presented. An example analysis of a monophonic signal is provided in Section 6.4.

6.2 Phase relations of harmonic partials

6.2.1 Concept

Pitched musical instruments generally produce harmonic sounds which can be represented by a superposition of P sinusoids at integer multiples of a fundamental frequency. Each harmonic partial can be described by its angular frequency $\omega_p > 0$, its amplitude $a_p \geq 0$ and an absolute phase shift $\varphi_p \in [-\pi, \pi)$: $\forall t \in \mathbb{Z}$,

$$s(t) = \sum_{p=1}^{P} a_p e^{j[\omega_p t + \varphi_p]}. \tag{6.1}$$

For perfectly harmonic sounds, the frequency of each harmonic is given as the p-th multiple of the fundamental frequency: $\omega_p = p\,\omega_1$. Complex exponentials are used here rather than real-valued cosine functions to reflect the fact that we only consider the baseband of the DFT in our model below.

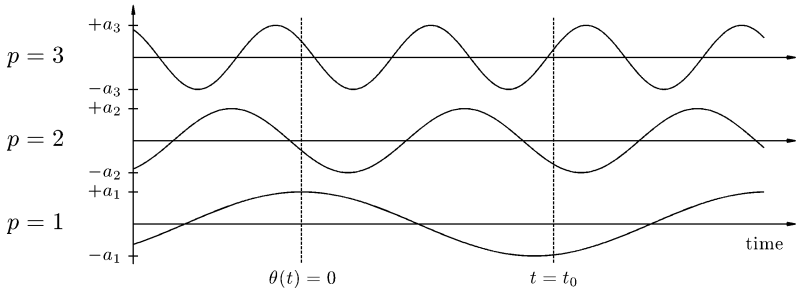
We are interested in the relations of the absolute phase shifts of the harmonic partials, that is, the way the partials are translated against each other along the time axis. To capture this relation, the phase shift of each partial is expressed in relation to the instantaneous phase of the fundamental frequency ω_1:

$$s(t) = \sum_{p=1}^{P} a_p e^{j[p\cdot\theta(t) + \Delta\varphi_p]}, \tag{6.2}$$

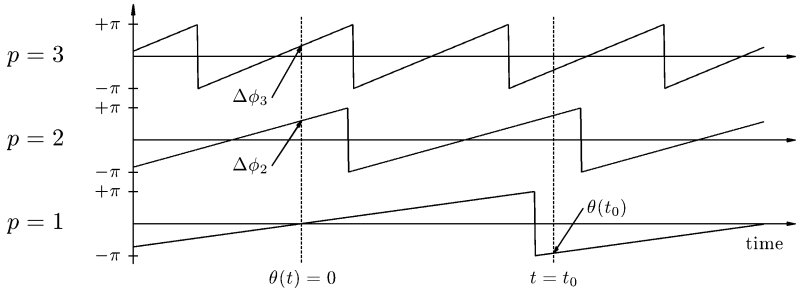
where $\theta(t) = \omega_1 t + \varphi_1$ denotes the instantaneous phase of the fundamental and $\Delta\varphi_p = \varphi_p - p\,\varphi_1$ represents the phase offset between the p-th partial and the fundamental (with $\Delta\varphi_1 = 0$).

Figure 6.1 shows a graphical illustration of the parameters in Eq. (6.2). The upper part (Fig. 6.1a) displays the waveform of the first three partials of a harmonic sound, and the lower part (Fig. 6.1b) the instantaneous phases. The phase offset $\Delta\varphi_p$ corresponds to the instantaneous phase of the partial at the time where $\theta(t) = 0$. Modifying $\Delta\varphi_p$ translates the p-th partial relative to the fundamental along the time axis. Since a translation by $\Delta\varphi_p$ is equivalent to a translation by $\Delta\varphi_p + c \cdot 2\pi$ with $c \in \mathbb{Z}$, $\Delta\varphi_p$ is uniquely defined in the range $[-\pi, \pi)$. Given all phase offsets $\Delta\varphi_p$ of the partials, the instantaneous phase of each partial can be computed at any given time t_0 based on the instantaneous phase of the fundamental $\theta(t_0)$ at that time. Note that even though we can

only measure the *wrapped* phase of $\theta(t)$ (i. e. in the interval $[-\pi, \pi)$), the correct wrapped phase of each partial can still be calculated.



(a) Waveform of the first three harmonics of a harmonic sound.

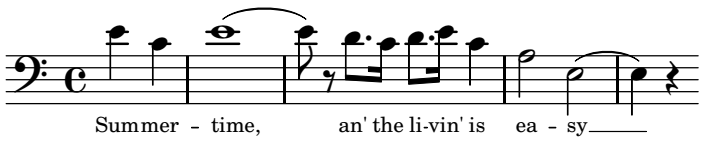


(b) Instantaneous phases of the first three harmonics of a harmonic sound.

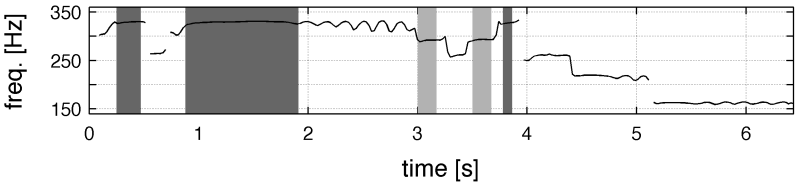
Figure 6.1: Illustration of the model parameters.

6.2.2 Example

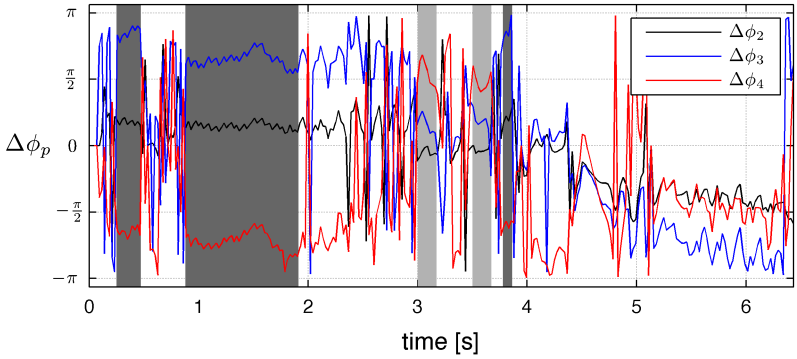
To illustrate the phase relations, the phase offsets $\Delta\varphi_p$ of the partials with indices $p \in \{2, 3, 4\}$ in an excerpt from a monophonic saxophone recording of 'Summertime' by G. Gershwin are displayed in Fig. 6.2. The score of the first four bars of this small excerpt is displayed in Fig. 6.2a, and Fig. 6.2b displays the fundamental frequency of the saxophone performance measured by the YIN algorithm (De Cheveigné and Kawahara, 2002). In Fig. 6.2c, the partial offsets $\Delta\varphi_p$ are plotted over time. Phase offsets were obtained from the STFT and computed as the wrapped difference between the measured instantaneous phases of each partial and p times the measured instantaneous phase of the fundamental. It can be seen that the partial offsets exhibit little variation during the steady state of each note — which is not surprising given the fact that the sound is harmonic. In addition to that, however, the same phase offsets occur at different note instances with the same pitch. The area shaded in dark grey highlights all renditions of the note E4 and the light grey area highlights all occurrences of the note D4. These observations make this property suitable for use in a



(a) First four bars of 'Summertime' by G. Gershwin.



(b) Fundamental frequencies of saxophone performance.



(c) Phase differences $\Delta\varphi_p$ over time.

Figure 6.2: Visualisation of the phase relations between the first four partials of a saxophone. The dark grey areas highlight all occurrences of the note E4, the light gray shaded areas all occurrences of the note D4. The diagram shows that the phase relations of a note are roughly constant over time and the phase relations are similar for repeated instances of notes with the same pitch.

complex matrix decomposition framework as we will illustrate in the next section. It should be noted that the relative phase offsets can only be defined if the partial frequencies are in a harmonic relation. For instruments with inharmonic frequency relations — such as the piano — a constant phase offset does not exist.

6.3 Parameter estimation

6.3.1 Frequency domain model

The parameters of the model in Eq. (6.2) are estimated from the STFT which is given by

$$X(n,k) = \sum_{t=-K}^{K-1} x(t+n\cdot m)\cdot h(t)\cdot e^{-j\Omega_k t}, \tag{6.3}$$

where $x(t)$ is the signal under analysis and n and k represent the time frame and frequency index, respectively. $h(t)$ denotes the analysis window of time support $[-K\ldots K-1]$. The distance between consecutive audio frames in samples (hop size) is denoted by m. $\Omega_k = \frac{2\pi k}{2K}$ is the normalised angular frequency of the k-th frequency index.

The STFT of the signal $s(t)$ from Eq. (6.2) is given by

$$S(n,k) = \sum_{p=1}^{P} a_p H(\Omega_k - p\omega_1) e^{j[p\Theta(n)+\Delta\varphi_p]}. \tag{6.4}$$

In this equation, $H(\Omega) = \sum_{t=-K}^{K-1} h(t)\cdot e^{-j\Omega t}$ denotes the Fourier spectrum of the window function $h(t)$ and $\Theta(n) = \theta(n\cdot m)$. A derivation of this equation can be found in Appendix D.1.

To simplify the monophonic model in Eq. (6.4), it is assumed that each partial can be represented by the *main lobe* of the window function only. This assumption holds fairly well if the side lobe attenuation of the window spectrum $H(\Omega)$ is sufficiently high and if the frequency resolution of the STFT is high enough so that the main lobes of adjacent partials do not overlap. The partial index belonging to frequency bin k is denoted by p_k. By setting $p_k = 0$ for all k that lie outside the main lobes of the partials, we ensure that these frequency bins are not assigned to any partial p. Additionally, we set $a_0 = 0$. This allows us to drop the sum over p in Eq. (6.4):

$$S'(n,k) = a_{p_k} H(\Omega_k - p_k\omega_1) e^{j[p_k\Theta(n)+\Delta\varphi_{p_k}]}. \tag{6.5}$$

In addition, a real time-varying gain factor $g(n) > 0$ is introduced that enables a uniform scaling of the magnitudes in order to accommodate loudness variations (similar to the gains in NMF):

$$\hat{B}(n,k) = g(n)\cdot S'(n,k). \tag{6.6}$$

Scaling ambiguities between $g(n)$ and a_p can be resolved by normalising a_p. Finally, the model can be extended to incorporate *multiple* harmonic sounds. We denote the index of each harmonic sound by r and append it to the quantities in Eq. (6.6):

$$\hat{V}(n,k) = \sum_{r=0}^{R-1} g_r(n) \cdot S'_r(n,k) \tag{6.7}$$

By substituting Eq. (6.5) into Eq. (6.7), we finally obtain:

$$\hat{V}(n,k) = \sum_{r=0}^{R-1} \underbrace{a_{p_{k,r}} H(\Omega_k - p_{k,r}\omega_{1,r}) e^{j\Delta\phi_{p_{k,r}}}}_{w_r(k)} \cdot \underbrace{g_r(n) e^{jp_{k,r}\Theta_r(n)}}_{h_r(n,k)} \tag{6.8}$$

$$= \sum_{r=0}^{R-1} w_r(k) \cdot h_r(n,k) \tag{6.9}$$

The term $w_r(k)$ is not time-dependent and is referred to as a *complex basis function*. Accordingly, the term $h_r(n,k)$ is referred to as a *complex activation*. Note that $h_r(n,k)$ is a 2-dimensional function. Equation (6.9) is therefore not a complex matrix *factorisation*, because it does not represent the complex spectrogram by a *matrix product*. But it is a *decomposition* of a complex spectrogram into a matrix of complex basis functions $w_r(k)$, a matrix of real-valued gain factors $g_r(n)$ and a matrix of real-valued instantaneous phases of the fundamentals $\Theta_r(n)$.

We will not investigate the case of multiple concurrent sounds in this thesis. The aim here is rather to prove that phase offsets between partials are a viable concept for sound analysis purposes. The model parameters will thus be estimated for the monophonic case only (Eq. (6.6)).

6.3.2 Parameter estimation

The parameters in Eq. (6.6) can be estimated by minimizing the error between the original complex spectrogram $B(n,k)$ and the model approximation $\hat{B}(n,k)$

for all $n \in [0 \dots N-1]$ and $k \in [0 \dots K-1]$ with $N \geq 0$ and $K \geq 0$. We choose to minimise the following cost function:

$$J = \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} \left| \ln(B(n,k)) - \ln(\hat{B}(n,k)) \right|^2 \tag{6.10}$$

$$= \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} \left[\ln \left(\frac{|B(n,k)|}{g(n) \cdot a_{p_k} H(\Omega_k - p_k \omega_1)} \right) \right]^2 + \left[\angle B(n,k) - \Delta\varphi_{p_k} - p_k \Theta(n) + 2\pi q(n,k) \right]^2 \tag{6.11}$$

where $\angle B(n,k)$ denotes the argument of the complex spectrogram $B(n,k)$. The term $q(n,k) \in \mathbb{Z}$ stems from the fact that the logarithm of a complex number has an infinite number of solutions which are obtained by adding integer multiples of 2π to the imaginary part of the solution (Sarason, 2007). The integer $q(n,k)$ is here treated as an additional parameter that has to be estimated. In Eq. (6.11), $H(\Omega)$ is assumed positive, since we only consider the main lobe of the window function. The model parameters are estimated by means of a coordinate descent (J is minimized w.r.t. each parameter):

$$g(n) = \left(\prod_{k=0}^{K} \frac{|B(n,k)|}{a_{p_k} H(\Omega_k - p_k \omega_1)} \right)^{\frac{1}{K}} \tag{6.12}$$

$$a_p = \left(\prod_{n=1}^{N} \prod_{\{k|p_k=p\}} \frac{|B(n,k)|}{g(n) H(\Omega_k - p\omega_1)} \right)^{\frac{1}{N \cdot \#\{k|p_k=p\}}} \tag{6.13}$$

$$\Theta(n) = \frac{\sum_{k=1}^{K} p_k [\angle B(n,k) - \Delta\varphi_{p_k} + 2\pi q(n,k)]}{\sum_{k=1}^{K} p_k^2} \tag{6.14}$$

$$\Delta\varphi_p = \frac{\sum_{n=1}^{N} \sum_{\{k|p_k=p\}} \angle B(n,k) - p\Theta(n) + 2\pi q(n,k)}{N \cdot \#\{k|p_k=p\}} \tag{6.15}$$

$$q(n,k) = \left\lfloor -\frac{1}{2\pi} \left[\angle B(n,k) - \Delta\varphi_{p_k} - p_k \Theta(n) \right] \right\rceil \tag{6.16}$$

In these equations, the expression $\{k|p_k = p\}$ represents the set of frequency indices k at which $p_k = p$, the operator $\#\{\dots\}$ denotes the cardinality of the set and $\lfloor \dots \rceil$ rounds a real number to the nearest integer. A derivation of these equations can be found in Appendix D.2.

6.4 Analysis of an example signal

This section illustrates how the estimation method can be used for a user-assisted transcription task. In a similar way as in Chapter 3, user input was employed

to guide the extraction of the parameters of the instrument model from the recording. The method was applied to an example signal, which consisted of the same monophonic saxophone recording of Summertime that was used to illustrate the phase relations in Fig. 6.2. It has a sample rate of 44.1 kHz and the first eight bars of the recording are used here.

Learning the parameters

To illustrate the application, the recording was split into two parts. It was assumed that the note labels were provided by the user for the first part of the recording, which consisted of the first four bars (cf. Fig. 6.2a). From these note labels, prototypical partial amplitudes a_p and phase offsets $\Delta\varphi_p$ were learned for the different pitches ω_1 in the following way: A spectrogram with $K = 2049$ frequency bins and $N = 5380$ time frames was computed. Note labels were required in order to segment the spectrogram in time into the different notes. In this single instrument scenario the labels could either be provided by a user or automatically extracted by means of a monophonic pitch estimation algorithm. All spectrogram parts with the same nominal pitch were concatenated, the fundamental frequency was estimated by employing the YIN algorithm and the average fundamental frequency ω_1 was computed for all pitches. $g(n)$ was estimated from the original spectrogram by taking the mean of the magnitudes in each time frame. In order to compute a_p, Eqs. (6.13) and (6.12) were alternatingly applied for 10 iterations. For the computation of $\Delta\varphi_p$, $\Theta(n)$ was initialised by the instantaneous phase value of the frequency bin corresponding to the fundamental in each frame. An initial estimate for $\Delta\varphi_p$ was obtained by replacing the terms in the summation in the numerator of Eq. (6.15) by the wrap function, which results in the following equation:

$$\Delta\varphi_p = \frac{\sum_{n=0}^{N-1} \sum_{\{k|p_k=p\}} \operatorname{wrap}(\angle B(n,k) - p\Theta(n))}{N \cdot \#\{k|p_k = p\}}, \tag{6.17}$$

where $\operatorname{wrap}(\alpha)$ calculates the principal argument of α. $q(n,k)$ was computed according to Eq. (6.16) and Eqs. (6.14)–(6.16) were iteratively applied until $q(n,k)$ converged. $\Delta\varphi_p$ was eventually given by the result of Eq. (6.15) in the last iteration.

The learned prototype amplitudes a_p and $\Delta\varphi_p$ were employed to estimate $g(n)$ and $\Theta(n)$ in the second part of the recording. The second part consisted of the remaining four bars of the Summertime example (Fig. 6.3a). First, $\Theta(n)$ was initialised with the instantaneous phase values at the frequency bins

corresponding to ω_1. Then $q(n,k)$ was estimated according to Eq. (6.16). Finally, $g(n)$ and $\Theta(n)$ were estimated according to Eqs. (6.12) and (6.14).

Activity detection

Active pitches can be estimated from both $g(n)$ and $\Theta(n)$. While for $g(n)$ this is obvious — high values indicate activity, low values indicate inactivity —, the instantaneous phase $\Theta(n)$ of the fundamental can also be used as an activity detector. We here used a measure inspired by the phase-based onset detection function described by Bello et al. (2005). The measure is based on the unwrapped phase, which can be assumed to be linear when the note is active and non-linear when the note is inactive. The unwrapped phase of $\Theta(n)$ will be denoted by $\Theta_u(n)$. The *second phase difference* can be used as a measure of phase-linearity. It is given by

$$\Delta\Theta_u(n) = \Theta_u(n) - 2\Theta_u(n-1) + \Theta_u(n-2). \tag{6.18}$$

If the unwrapped phase is strictly linear, $\Delta\Theta_u(n)$ will be close to zero, if it is non-linear $\Delta\Theta_u(n)$ is likely to take on values with larger magnitudes. Additionally, $\Delta\Theta_u(n)$ is likely to take on low values in several *consecutive active* frames and more random values in *consecutive inactive* frames. We therefore computed the mean square of $\Delta\Theta_u(n)$ over a sliding window as

$$\sigma(n) = \frac{1}{L} \sum_{n'=-\lfloor \frac{L}{2} \rceil}^{\lfloor \frac{L}{2} \rceil - 1} \Delta\Theta_u^2(n+n'), \tag{6.19}$$

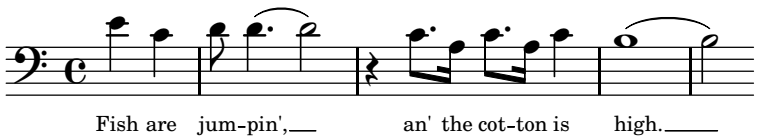
and defined the phase-based activity measure as

$$f(n) = -\ln(\sigma(n))\,. \tag{6.20}$$

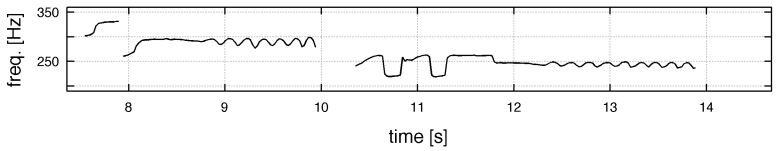
In our simulations a window length of 50 ms ($L = 37$) was empirically chosen.

Results

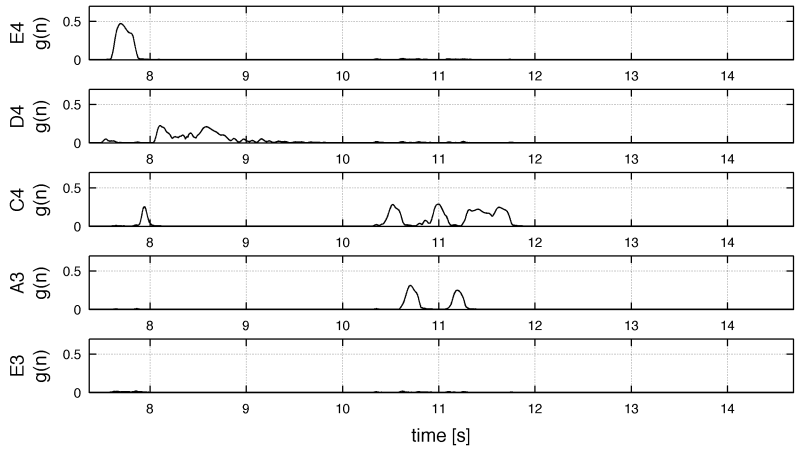
The results of the estimation are displayed in Fig. 6.3. In Fig. 6.3b, the measured fundamental frequencies of the four bar excerpt in Fig. 6.3a are shown. Figure 6.3c shows the gains $g(n)$ and Fig. 6.3d the results for the phase-based activity measure $f(n)$. The gains clearly show the activity of the different pitches and are very much reminiscent of activity measurements in NMF analyses. The results of the phase-based activity measure also reveal the active pitches very well, which confirms that the phase relations between harmonic partials can actually be used to characterise pitches of certain instruments and distinguish between them.



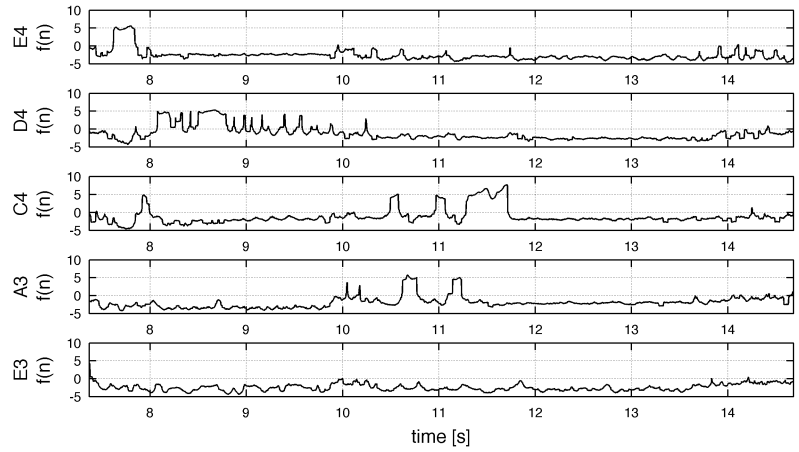
(a) Bars 5–8 of 'Summertime' by G. Gershwin.



(b) Fundamental frequencies for bars 5–8.



(c) Gains $g(n)$ for bars 5–8 for each pitch from bars 1–4.



(d) Phase activations $f(n)$ for bars 5–8 for each pitch from bars 1–4.

Figure 6.3: Example analysis of a monophonic saxophone example.

An examination of the estimated phase offsets $\Delta\varphi_p$ showed a good convergence towards the measured phase offsets from the spectrogram. Note that the pitch E3 does not occur in bars 5–8, and that the note B3, the last note in Fig. 6.3a, is missing because it did not occur in bars 1–4.

6.5 Summary and discussion

In this chapter the relative phase relations within the sustained part of harmonic sounds and their potential use for complex matrix decomposition were investigated. The phase relations between harmonic partials were expressed as relative phase offsets of the partials from the fundamental. Equations for the estimation of the model parameters have been presented based on a complex logarithmic cost function between the original spectrogram and the model approximation. With the analysis of an example signal, it was demonstrated that the relative phase offsets can be used as recurring, time-invariant phase characteristics of harmonic sounds.

As mentioned in Section 6.2.2, harmonic partial relations are a necessary condition in order to ensure constant phase relations. Bowed string instruments, as well as brass and reed instruments meet this condition which is caused by *mode locking* (Fletcher, 1978). If the partial relations are inharmonic, the condition of constant phase relations is not fulfilled. In particular, struck and plucked string instruments such as the piano and the guitar have inharmonic spectra and are therefore not suited for this concept. For harmonic sounds it has to be investigated to what extent the phase relations recur when the instrument plays several notes at the same pitch. Only when this condition is met is it possible to use the phase relations in the proposed way.

In future work the method should be extended to deal with *mixtures* of harmonic sounds in order to obtain a complex matrix decomposition that can be used to identify the underlying spectral components in the complex domain. A formulation of such a complex matrix decomposition framework has been provided in Section 6.3.1. For the monophonic case, the complex logarithmic cost function proved to be useful, not only because logarithmic amplitudes match human perception better than linear amplitudes, but also because it separates the modulus and argument of the model, which allowed us to treat them separately. In the polyphonic case however, a complex matrix decomposition framework would need to deal with magnitudes and phases jointly, since the sum of two complex time-frequency components depends on both their moduli and phases.

Chapter 7

Conclusion

The relatively long history of research on automatic music transcription and the fact that the accuracies of current state-of-the-art AMT systems still lag behind those achieved by human experts provoke the question in what ways a human user can aid the automatic transcription process. In this thesis we looked at ways to involve a musically-trained human user in the transcription process. The focus was mainly on employing user information to build instrument models which can be used for multi-instrument transcription. In the following section we will summarise the thesis and highlight the main results and contributions. In Section 7.2, several future directions for both user-assisted and fully automatic transcription systems will be proposed. We conclude the thesis with some closing remarks in Section 7.3.

7.1 Summary of contributions

We started our investigation on user-assisted music transcription by defining criteria and practical constraints for user requests in Chapter 2 and identified potential types of information that could be provided by a musically-trained user. Not all types of information appear to be equally useful from an algorithmic point of view and we therefore focussed on information that enables the use of timbre models for the instruments in the recording in order to enable the transcription of the individual instrument parts in multi-instrument recordings.

In Chapter 3, two different types of user information and their corresponding timbre models were compared. Providing the identities of the instruments in the recording enables the use of generic timbre models from an instrument sound database, while note annotations by the user allow us to extract instrument models for the specific instruments in the recording. In order to compare the performance of the different timbre models, a set of metrics was proposed

that enables the evaluation of the quality of pitch activation functions without requiring explicit decisions about the pitches present in each time frame. The results of our comparison of the two types of models showed that specific instrument templates can lead to considerably better pitch activation functions than generic templates when a large number of note annotations were provided by the user. Following on from this result, a method was proposed that enabled the extraction of multiple spectral templates per instrument and pitch within a non-negative analysis framework. Our evaluation showed that this method increased the accuracy of the gain matrices by a few percentage points.

Chapter 4 further investigated the timbre models based on note annotations and looked at ways of limiting the number of annotations a user has to provide. A limited set of note annotations will lead to incomplete timbre models in which no examples are provided for some pitches. In this case, the templates have to be estimated from the set of extracted spectra. Several methods for the estimation of missing templates were experimentally compared. A source-filter model was developed that is capable of estimating a non-white excitation spectrum. Furthermore a method was proposed to adapt a complete set of generic spectra to the spectra of a specific instrument of the same type by estimating a common adaptation filter curve. The results of our experiments showed that among the investigated estimation methods, the data-driven methods of copying and interpolating achieved the highest accuracies, particularly when note labels were provided at high pitch resolutions. The method of adapting previously extracted database templates was not strongly affected by different pitch resolutions of the note annotations and the source-filter model provided the least accurate results.

The process of grouping pitch activations of a multi-instrument recording into instrument streams was addressed in Chapter 5. A method was proposed to track the pitches over time. The method consisted of three processing stages: the computation of an instrument-independent pitch activation function, the formation and selection of candidate assignments of pitches to instruments, and finally the actual pitch tracking by means of the Viterbi algorithm. The transition probability between the different pitch-instrument candidates in the Viterbi algorithm was based on three different criteria: the reconstruction error of the candidate, the pitch continuity across frames and the activity status of each instrument. The method outperformed other multi-instrument pitch tracking methods.

In Chapter 6, the extraction of instrument models that include phase information was investigated as a step towards complex matrix decomposition. The relative phase offsets between the partials of harmonic sounds were explored and formally defined. Equations for the estimation of magnitude and phase parameters of the model were derived for the monophonic case and the application of

the model for a user-assisted transcription task was illustrated with a saxophone example.

7.2 Future directions

While carrying out the work for this theses, several ideas for future directions in this research area came to mind, which are detailed below.

Employing other types of user information

In this work we have only started to explore the numerous ways in which users can be engaged in the transcription process. Employing user information for the extraction of timbre models of the instruments appeared useful for the initial low-level analysis of music recordings. Other types of user information and interactions, however, might also be of benefit and have yet to be investigated. A few examples have been provided in Section 2.3.2. It is important that the information is well represented in the computational model which poses a challenge for certain types of high-level user information. A model of musical context has been proposed for chord recognition (Mauch, 2010) that might equally be useful for note transcription and might be able to incorporate more high-level information about the music to be analysed.

Tailoring latent variable models to music spectrogram data

Latent variable models such as NMF or PLCA for transcription are often employed by representing the average spectrum of a note by a *single* spectral template. This assumes that the set of short-time spectra at a certain pitch are more or less clustered around the spectral template so that the reconstruction error is small when the short-time spectra are represented by that single template. Prior research has shown that the short-time spectra of a note rather form a trajectory than a cluster, or at least a combination of a trajectory and a cluster (Burred and Röbel, 2010). In Section 3.4 we proposed an algorithm to extract several spectral templates as a step towards a more flexible representation. Virtanen (2007) enforced the temporal continuation of templates, but did not allow a modification of their shape. Benetos and Dixon (2011a) modelled different sound states by integrating HMMs and latent variable models assuming a fixed succession of sound states. All these approaches require prior knowledge about the instruments. It might, however, be possible to utilise the trajectorial nature of short-time spectra to formulate a non-parametric model for detecting latent note events in polyphonic music.

Building more accurate and specific instrument models

The results of the comparison of the estimation methods for missing spectral templates revealed that none of the more advanced instrument models — the source-filter model and the database spectra adaptation method — was able to achieve a very accurate representation of the instruments. Particularly the source-filter model proved to be not very well suited to model the average spectral shapes at different pitches for certain instrument types. By visualising the average spectral shapes of isolated recorded notes of some instruments, it becomes obvious that the source-filter model even with our proposed non-white excitation spectrum is a clear oversimplification of the real spectra. Nevertheless this model has been used in various contexts of music analysis and is still considered the most generic available instrument model. Our experiments showed, however, that inaccurate spectral templates estimated by the source-filter model considerably affect the accuracy of the initial pitch analysis. This could be remedied by more specific instrument models that incorporate the physical properties of the instruments and enable an adaptation of instrument specific parameters. Parametric physical models might also help to considerably reduce the number of parameters to be estimated.

Exploring phase properties of musical instruments

The importance of modelling the phase spectrum of musical instruments has been highlighted in Section 6.1. Phase information is very often neglected in the research area of computational music analysis. A Fourier transform of length N of a temporal segment with the same number of audio samples results in $\frac{N}{2}$ unique magnitude bins and $\frac{N}{2}$ phase bins. Disregarding the phase spectrum hence reduces the information content by 50%. Our attempts to make use of the phase relations between harmonic partials show that it is possible to make use of certain properties of the phase. This property could be further explored and its validity for different instruments and instrument groups needs to be investigated and explained in terms of the various sound production mechanisms of musical instruments. Another more generic property of the phase is its ability to link sinusoidal partials across frames which can provide useful hints for partial tracking and note tracking.

Including phase information in latent variable models

Finally, it would be useful to include phase properties in latent variable models and thus to enable these models to explain the complex spectrogram of music signals as opposed to just the magnitudes. The concept of non-negativity still holds for the modulus of complex sinusoids, but it needs to be extended to model

the phase angle and the fact that this angle can only be measured in the interval $[-\pi, \pi)$. Our proposed complex matrix decomposition approach provides one way of doing this which still has to be extended for the multi-instrument case.

7.3 Closing remarks

User-assisted music transcription systems are useful tools and have potential applications in a range of different areas. Particularly musicological research would benefit from such systems in different ways: by facilitating the notation of previously unnotated recorded music such as traditional folk music, tribal music or jazz solos, and by enabling performance studies that analyse parameters such as vibrato and intonation in ensemble recordings which would otherwise be inaccessible for musicologists. Practising musicians and arrangers would benefit from such systems by obtaining transcriptions in a shorter time, and audio engineers as well as music producers might use the transcription results to control audio effects such as harmonisers or synthesisers.

The idea of building machines that perform audio analysis tasks completely autonomically and with results that exceed those of human experts is very intriguing. The results of the annual MIREX evaluation in recent years have shown, however, that rates of improvement for various tasks have slowed down and the term *glass ceiling* appears more often. The development of user-assisted analysis systems can be a means to spur the research of these tasks. Being able to access reliable information from a user for specific subtasks of the transcription process allows algorithmic development to focus on the remaining subtasks. This fosters more detailed evaluations of individual subtasks which might give useful insights that could be beneficial for fully automatic transcription systems as well. All this will hopefully help towards a better understanding of the underlying acoustical processes of music performance and recording.

With this work we have only scratched the surface of what is actually possible when humans and computers collaborate on audio analysis tasks such as music transcription. We hope that this work inspires further research in this direction that helps to build more accurate, robust and versatile transcription systems and to improve the audio analysis workflow by human/computer cooperation.

Appendix A

Derivations of update equations for the non-negative analysis framework

The mathematical derivations of the multiplicative update rules for the non-negative analysis framework in Section 3.2 are presented here. Section A.1 derives the update equations for the basis function matrices $\mathbf{W}^{\phi,i}$; the equations for the gain matrices $\mathbf{H}^{\phi,i}$ are derived in Section A.2. The derivations in both sections are structured in the following way: First, the cost function is differentiated w.r.t. to the matrix elements of $\mathbf{W}^{\phi,i}$ and $\mathbf{H}^{\phi,i}$, respectively. Second, gradient descent is applied to those elements. In this step, the learning rate is chosen in such a way that the update equations can be expressed by simple multiplicative rules.

The mixture model from Eq. (3.1) is given in its *elementwise* form by

$$\mathbf{V}_{k,n} \approx \mathbf{\Lambda}_{k,n} = \sum_{i=0}^{I-1} \sum_{\phi=0}^{\Phi-1} \sum_{r=0}^{R-1} \mathbf{W}^{\phi,i}_{k-\phi,r} \mathbf{H}^{\phi,i}_{r,n}. \tag{A.1}$$

In this equation, the subscripts refer to the row and column indices of the matrices.

The cost function (see Eq. (2.9)) computes the β-divergence between the matrix elements of the original spectrogram and its approximation:

$$C_\beta = \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} d_\beta(\mathbf{V}_{k,n}, \mathbf{\Lambda}_{k,n}). \tag{A.2}$$

The β-divergence is given by (see Eq. (2.5))

$$d_\beta(x,y) = \frac{x^\beta}{\beta(\beta-1)} + \frac{y^\beta}{\beta} - \frac{xy^{\beta-1}}{\beta-1}, \tag{A.3}$$

for $\beta \in \mathcal{R}\backslash\{0,1\}$, and

$$d_0(x,y) = d_{IS}(x,y) = \frac{x}{y} - \log\left(\frac{x}{y}\right) - 1 \tag{A.4}$$

$$d_1(x,y) = d_{KL}(x,y) = x \cdot \log\left(\frac{x}{y}\right) + y - x. \tag{A.5}$$

It enables the use of various well-known cost-functions, such as the least squares error, the Kullback-Leibler divergence and the Itakura-Saito divergence.

A.1 Update equations for $\mathbf{W}^{\phi,i}$

We first derive an auxiliary term, which we will use for the differentiation of the cost function C_β. This auxiliary term is the differentiation of $\mathbf{\Lambda}$ w.r.t. the matrix elements of $\mathbf{W}^{\phi,i}$:

$$\begin{aligned}
\frac{\partial \mathbf{\Lambda}_{k',n}}{\partial \mathbf{W}^{\phi,i}_{k,r}} &= \frac{\partial}{\partial \mathbf{W}^{\phi,i}_{k,r}} \left(\sum_{i'=0}^{I-1} \sum_{\phi'=0}^{\Phi-1} \sum_{r'=0}^{R-1} \mathbf{W}^{\phi',i'}_{k'-\phi',r'} \mathbf{H}^{\phi',i'}_{r',n} \right) \\
&= \frac{\partial}{\partial \mathbf{W}^{\phi,i}_{k,r}} \left(\mathbf{W}^{\phi,i}_{k'-\phi,r} \mathbf{H}^{\phi,i}_{r,n} \right) \\
&= \begin{cases} \mathbf{H}^{\phi,i}_{r,n} & \text{if } k'-\phi = k \\ 0 & \text{otherwise} \end{cases}.
\end{aligned} \tag{A.6}$$

Given this result, we derive the gradient of the cost function C_β w.r.t. the elements of $\mathbf{W}^{\phi,i}$:

$$\begin{aligned}
\frac{\partial C_\beta}{\partial \mathbf{W}^{\phi,i}_{k,r}} &= \frac{\partial C_\beta}{\partial \mathbf{\Lambda}_{k',n}} \cdot \frac{\partial \mathbf{\Lambda}_{k',n}}{\partial \mathbf{W}^{\phi,i}_{k,r}} \\
&= \sum_{n=0}^{N-1} \mathbf{\Lambda}^{\beta-1}_{\phi+k,n} \mathbf{H}^{\phi,i}_{r,n} - \mathbf{V}_{\phi+k,n} \mathbf{\Lambda}^{\beta-2}_{\phi+k,n} \mathbf{H}^{\phi,i}_{r,n}.
\end{aligned} \tag{A.7}$$

The sum over k' has been dropped in this step, since we are dealing with a fixed ϕ and k which determines k' (cf. Eq. (A.6)).

The gradient descent in its general form is given by

$$\mathbf{W}_{k,r}^{\phi,i} \leftarrow \mathbf{W}_{k,r}^{\phi,i} - \eta_W \frac{\partial C_\beta}{\partial \mathbf{W}_{k,r}^{\phi,i}}, \tag{A.8}$$

where η_W denotes the learning rate. In order to obtain multiplicative update rules, η_W is selected non-uniformly for the matrix elements in the following way:

$$\eta_W = \frac{\mathbf{W}_{k,r}^{\phi,i}}{\sum_{n=0}^{N-1} \mathbf{\Lambda}_{\phi+k,n}^{\beta-1} \mathbf{H}_{r,n}^{\phi,i}}. \tag{A.9}$$

The update equation is then given by

$$\mathbf{W}_{k,r}^{\phi,i} \leftarrow \mathbf{W}_{k,r}^{\phi,i} \cdot \frac{\sum_{n=0}^{N-1} \mathbf{V}_{\phi+k,n} \mathbf{\Lambda}_{\phi+k,n}^{\beta-2} \mathbf{H}_{r,n}^{\phi,i}}{\sum_{n=0}^{N-1} \mathbf{\Lambda}_{\phi+k,n}^{\beta-1} \mathbf{H}_{r,n}^{\phi,i}} \tag{A.10}$$

and in matrix notation by

$$\mathbf{W}^{\phi,i} \leftarrow \mathbf{W}^{\phi,i} \bullet \frac{\left(\overset{\phi\uparrow}{\mathbf{V}} \bullet \mathbf{\Lambda}^{\overset{\phi\uparrow}{\beta-2}}\right) \left[\mathbf{H}^{\phi,i}\right]^\top}{\left(\mathbf{\Lambda}^{\overset{\phi\uparrow}{\beta-1}}\right) \left[\mathbf{H}^{\phi,i}\right]^\top}. \tag{A.11}$$

In this equation, $\bullet$ denotes elementwise multiplication and the division and power operations are also carried out elementwise. The operator $\phi\uparrow$ denotes an upward shift of the matrix elements by ϕ rows while the lower ϕ rows are filled with zeros.

A.2 Update equations for $\mathbf{H}^{\phi,i}$

For the derivation of the update equations we follow the same steps as in Section A.1. The auxiliary term in this case is given by

$$\begin{aligned}\frac{\partial \mathbf{\Lambda}_{k,n'}}{\partial \mathbf{H}^{\phi,i}_{r,n}} &= \frac{\partial}{\partial \mathbf{H}^{\phi,i}_{r,n}} \left(\sum_{i'=0}^{I-1} \sum_{\phi'=0}^{\Phi-1} \sum_{r'=0}^{R-1} \mathbf{W}^{\phi',i'}_{k-\phi',r'} \mathbf{H}^{\phi',i'}_{r',n'} \right) \\ &= \frac{\partial}{\partial \mathbf{H}^{\phi,i}_{r,n}} \left(\mathbf{W}^{\phi,i}_{k-\phi,r} \mathbf{H}^{\phi,i}_{r,n'} \right) \\ &= \begin{cases} \mathbf{W}^{\phi,i}_{k-\phi,r} & \text{if } n = n' \\ 0 & \text{otherwise} \end{cases} . \end{aligned} \tag{A.12}$$

The gradient of the cost function w.r.t. $\mathbf{H}^{\phi,i}$ is given by

$$\begin{aligned}\frac{\partial C_\beta}{\partial \mathbf{H}^{\phi,i}_{r,n}} &= \frac{\partial C_\beta}{\partial \mathbf{\Lambda}_{k,n'}} \cdot \frac{\partial \mathbf{\Lambda}_{k,n'}}{\partial \mathbf{H}^{\phi,i}_{r,n}} \\ &= \sum_{k=0}^{K-1} \mathbf{\Lambda}^{\beta-1}_{k,n} \mathbf{W}^{\phi,i}_{k-\phi,r} - \mathbf{V}_{k,n} \mathbf{\Lambda}^{\beta-2}_{k,n} \mathbf{W}^{\phi,i}_{k-\phi,r}. \end{aligned} \tag{A.13}$$

In the equation for the gradient descent

$$\mathbf{H}^{\phi,i}_{r,n} \leftarrow \mathbf{H}^{\phi,i}_{r,n} - \eta_H \frac{\partial C_\beta}{\partial \mathbf{H}^{\phi,i}_{r,n}} \tag{A.14}$$

we choose

$$\eta_H = \frac{\mathbf{H}^{\phi,i}_{r,n}}{\sum_{k=0}^{K-1} \mathbf{\Lambda}^{\beta-1}_{k,n} \mathbf{W}^{\phi,i}_{k-\phi,r}}. \tag{A.15}$$

This yields the update equation in its elementwise form:

$$\mathbf{H}^{\phi,i}_{r,n} \leftarrow \mathbf{H}^{\phi,i}_{r,n} \cdot \frac{\sum_{k=0}^{K-1} \mathbf{V}_{k,n} \mathbf{\Lambda}^{\beta-2}_{k,n} \mathbf{W}^{\phi,i}_{k-\phi,r}}{\sum_{k=0}^{K-1} \mathbf{\Lambda}^{\beta-1}_{k,n} \mathbf{W}^{\phi,i}_{k-\phi,r}}, \tag{A.16}$$

and in matrix notation

$$\mathbf{H}^{\phi,i} \leftarrow \mathbf{H}^{\phi,i} \bullet \frac{\left[\overset{\phi\downarrow}{\mathbf{W}^{\phi,i}} \right]^\top \left(\mathbf{V} \bullet \mathbf{\Lambda}^{\beta-2} \right)}{\left[\overset{\phi\downarrow}{\mathbf{W}^{\phi,i}} \right]^\top \mathbf{\Lambda}^{\beta-1}}. \tag{A.17}$$

Appendix B

Derivations and technical details for the source-filter model

B.1 Update equations

The source-filter model from Eq. (4.1) is given in its elementwise form by

$$\hat{W}_{k,d} = s_d \cdot e_{k-\phi_d} \cdot h_k \tag{B.1}$$

In order to obtain the multiplicative update equations for the source-filter model, gradient descent is applied to each of the three terms s_d, e_k and h_k, and the step width is chosen in such a way that the update equation becomes multiplicative. In all cases, the beta divergence $C_\beta\left(\mathbf{W}', \hat{\mathbf{W}}'\right)$ is used as a cost function to measure the deviation between the estimated spectra and the original spectra (cf. Section 2.4.1).

B.1.1 Scaling factors s

We derive the update equations for the scaling factors s_d by applying gradient descent to the cost function. The gradient of the beta divergence w.r.t. s_d is given by:

$$\frac{\partial C_\beta(\mathbf{W}', \hat{\mathbf{W}}')}{\partial s_d} = \frac{\partial}{\partial s_d} \sum_{k=0}^{K-1} \sum_{d'=0}^{D-1} \frac{{W'}_{k,d'}^{\beta}}{\beta(\beta-1)} + \frac{{\hat{W}'}_{k,d'}^{\beta}}{\beta} - \frac{{W'}_{k,d'} {\hat{W}'}_{k,d'}^{\beta-1}}{\beta-1} \tag{B.2}$$

$$= \sum_{k=0}^{K-1} {\hat{W}'}_{k,d}^{\beta-1} \cdot \frac{\partial \hat{W}'_{k,d}}{\partial s_d} - {W'}_{k,d} {\hat{W}'}_{k,d}^{\beta-2} \cdot \frac{\partial \hat{W}'_{k,d}}{\partial s_d} \tag{B.3}$$

$$= \sum_{k=0}^{K-1} {\hat{W}'}_{k,d}^{\beta-1} e_{k-\phi_d} h_k - {W'}_{k,d} {\hat{W}'}_{k,d}^{\beta-2} e_{k-\phi_d} h_k. \tag{B.4}$$

Note that $\mathbf{W}'$ does not depend on s_d which causes the first term under the sum in Eq. (B.2) to disappear. Likewise the sum over d' vanishes since $\hat{W}'_{k,d'}$ only depends on s_d if $d' = d$.

The gradient descent update equation for s_d in its generic form is given by

$$s_d \leftarrow s_d - \eta_s \frac{\partial C_\beta(\mathbf{W}', \hat{\mathbf{W}}')}{\partial s_d}. \tag{B.5}$$

In order to obtain multplicative update equations, we set

$$\eta_s = \frac{s_d}{\sum_{k=0}^{K-1} {\hat{W}'}_{k,d}^{\beta-1} e_{k-\phi_d} h_k}, \tag{B.6}$$

and obtain the following update rule for s_d

$$s_d \leftarrow s_d \cdot \frac{\sum_{k=0}^{K-1} {W'}_{k,d} {\hat{W}'}_{k,d}^{\beta-2} e_{k-\phi_d} h_k}{\sum_{k=0}^{K-1} {\hat{W}'}_{k,d}^{\beta-1} e_{k-\phi_d} h_k}. \tag{B.7}$$

B.1.2 Excitation spectrum e

Update equations for the excitation spectrum are derived in the same way as in the previous section. In order to derive the partial derivative of $C_\beta(\mathbf{W}', \hat{\mathbf{W}}')$ w.r.t. e_k, we start by differentiating the β-divergence w.r.t. $e_{k'-\phi_d}$:

$$\frac{\partial C_\beta(\mathbf{W}', \hat{\mathbf{W}}')}{\partial e_{k'-\phi_d}} = \frac{\partial}{\partial e_{k'-\phi_d}} \sum_{d'=0}^{D-1} \sum_{k''=0}^{K-1-\phi_d} \frac{{W'}^{\beta}_{k'',d'}}{\beta(\beta-1)} + \frac{\left(s_{d'} \cdot e_{k''-\phi_{d'}} \cdot h_{k''}\right)^\beta}{\beta} - \frac{{W'}_{k'',d'} \left(s_{d'} \cdot e_{k''-\phi_{d'}} \cdot h_{k''}\right)^{\beta-1}}{\beta-1}. \tag{B.8}$$

By substituting $k = k' - \phi_d$, we obtain

$$\frac{\partial C_\beta(\mathbf{W}', \hat{\mathbf{W}}')}{\partial e_k} = \frac{\partial}{\partial e_k} \sum_{d=0}^{D-1} \sum_{k'=k+\phi_d}^{K-1} \frac{{W'}^{\beta}_{k',d}}{\beta(\beta-1)} + \frac{\left(s_d \cdot e_k \cdot h_{k'}\right)^\beta}{\beta} - \frac{{W'}_{k',d} \left(s_d \cdot e_k \cdot h_{k'}\right)^{\beta-1}}{\beta-1} \tag{B.9}$$

$$= \sum_{d=0}^{D-1} \left(s_d \cdot e_k \cdot h_{k+\phi_d}\right)^{\beta-1} \cdot s_d \cdot h_{k+\phi_d} - {W'}_{k+\phi_d,d} \left(s_d \cdot e_k \cdot h_{k+\phi_d}\right)^{\beta-2} \cdot s_d \cdot h_{k+\phi_d} \tag{B.10}$$

$$= \sum_{d=0}^{D-1} {\hat{W}'}^{\beta-1}_{k+\phi_d,d} \cdot s_d \cdot h_{k+\phi_d} - {W'}_{k+\phi_d,\phi_d} {\hat{W}'}^{\beta-2}_{k+\phi_d,d} \cdot s_d \cdot h_{k+\phi_d}. \tag{B.11}$$

The gradient descent update equation is given by

$$e_k \leftarrow e_k - \eta_e \frac{\partial C_\beta(\mathbf{W}', \hat{\mathbf{W}}')}{\partial e_k}. \tag{B.12}$$

By choosing the learning rate as

$$\eta_e = \frac{e_k}{\sum_{d=0}^{D-1} {\hat{W}'}^{\beta-1}_{k+\phi_d,d} \cdot s_d \cdot h_{k+\phi_d}}, \tag{B.13}$$

we obtain the multiplicative update rule

$$e_k \leftarrow e_k \cdot \frac{\sum_{d=0}^{D-1} {W'}_{k+\phi_d,d} \cdot {\hat{W}'}^{\beta-2}_{k+\phi_d,d} \cdot s_d \cdot h_{k+\phi_d}}{\sum_{d=0}^{D-1} {\hat{W}'}^{\beta-1}_{k+\phi_d,d} \cdot s_d \cdot h_{k+\phi_d}}. \tag{B.14}$$

B.1.3 Filter response h

The update rules for $\mathbf{h}$ are obtained in the same fashion. Here, the gradient is given by

$$\frac{\partial C_\beta(\mathbf{W}', \hat{\mathbf{W}}')}{\partial h_k} = \frac{\partial}{\partial h_k} \sum_{d=0}^{D-1} \sum_{k'=0}^{K-1} \frac{{W'}_{k',d}^{\beta}}{\beta(\beta-1)} + \frac{{\hat{W}'}_{k',d}^{\beta}}{\beta} - \frac{{W'}_{k',d}{\hat{W}'}_{k',d}^{\beta-1}}{\beta-1} \tag{B.15}$$

$$= \sum_{\{d|\phi_d \leq k\}} {\hat{W}'}_{k,d}^{\beta-1} \cdot \frac{\partial \hat{W}'_{k,d}}{\partial h_k} - {W'}_{k,d}{\hat{W}'}_{k,d}^{\beta-2} \cdot \frac{\partial \hat{W}'_{k,d}}{\partial h_k} \tag{B.16}$$

$$= \sum_{\{d|\phi_d \leq k\}} {\hat{W}'}_{k,d}^{\beta-1} \cdot s_d \cdot e_{k-\phi_d} - {W'}_{k,d}{\hat{W}'}_{k,d}^{\beta-2} \cdot s_d \cdot e_{k-\phi_d}. \tag{B.17}$$

In the gradient descent update equation

$$h_k \leftarrow h_k - \eta_h \frac{\partial C_\beta(\mathbf{W}', \hat{\mathbf{W}}')}{\partial h_k}, \tag{B.18}$$

we set

$$\eta_h = \frac{h_k}{\sum\limits_{\{d|\phi_d \leq k\}} {\hat{W}'}_{k,d}^{\beta-1} \cdot s_d \cdot e_{k-\phi_d}}, \tag{B.19}$$

and obtain the multiplicative update rule

$$h_k \leftarrow h_k \cdot \frac{\sum\limits_{\{d|\phi_d \leq k\}} {W'}_{k,d} \cdot {\hat{W}'}_{k,d}^{\beta-2} \cdot s_d \cdot e_{k-\phi_d}}{\sum\limits_{\{d|\phi_d \leq k\}} {\hat{W}'}_{k,d}^{\beta-1} \cdot s_d \cdot e_{k-\phi_d}}. \tag{B.20}$$

B.2 Ambiguities

The model in Eq. (4.1) contains two ambiguities which need to be addressed in order to provide unique results for $\mathbf{s}$, $\mathbf{e}$ and $\mathbf{h}$.

B.2.1 Scaling

Scaling $\mathbf{e}$ and $\mathbf{h}$ by constant factors c_1 and c_2, and $\mathbf{s}$ by the inverse of the product of these factors, results in the same estimates of the spectra:

$$\hat{W}'_{k,d} = \frac{1}{c_1 c_2} s_d \cdot c_1 e_{k-\phi_d} \cdot c_2 h_k \tag{B.21}$$

$$= s_d \cdot e_{k-\phi_d} \cdot h_k \tag{B.22}$$

To fix this ambiguity, the vectors **e** and **h** can be scaled to unit length, which determines c_1 and c_2, and **s** can be multiplied by $\frac{1}{c_1 c_2}$ to compensate for that.

B.2.2 Multiplication by exponential function

The second ambiguity is given when **s**, **e** and **h** are multiplied by exponential functions with the same base α. More precisely, if s_d is multiplied by the function α^{ϕ_d}, $e_{k-\phi_d}$ by $\alpha^{k-\phi_d}$ and h_k by α^{-k}, the same spectra as in the original model are obtained:

$$\hat{W'}_{k,d} = \alpha^{\phi_d} s_d \cdot \alpha^{k-\phi_d} e_{k-\phi_d} \cdot \alpha^{-k} h_k \tag{B.23}$$

$$= s_d \cdot e_{k-\phi_d} \cdot h_k, \tag{B.24}$$

since $\alpha^{\phi_d} \cdot \alpha^{k-\phi_d} \cdot \alpha^{-k} = \alpha^0 = 1$. In other words, for any value of α the source-filter model yields the exact same spectra. Intuitively, this results in *tilts* of the vectors **s**, **e** and **h**, depending on the value of α: if $\alpha > 1$, **s** and **e** get tilted clockwise, and **h** anticlockwise, if $\alpha < 1$, the vectors are tilted the other way round, if $\alpha = 1$, none of the vectors is tilted.

In order to fix this ambiguity, one of the vectors **s**, **e** or **h** must be set to a specific tilt. This defines α and thus makes the solution unambiguous. We require here that the scaling factors **s** are not tilted at all. The scaling factors will exhibit different values for each pitch, but it would be unlikely that a systematical increase or decrease towards higher or lower pitches can be observed.

Mathematically, the no-tilt condition can be imposed by requiring that the differences between subsequent vector elements sum to 0. If we define **s** as a function of ϕ: $\mathbf{s} := s(\phi)$, and if we further assume that $s(\phi)$ can be expressed by

$$s(\phi) = s^*(\phi) \cdot \alpha^{\phi}, \tag{B.25}$$

where $s^*(\phi)$ is the untilted version of $s(\phi)$, we can impose the no tilt condition on $s^*(\phi)$ by enforcing the sum of all first derivatives of $s(\phi)$ to be zero:

$$\int_{\phi_{\min}}^{\phi_{\max}} \frac{ds^*(\phi)}{d\phi} d\phi = 0 \tag{B.26}$$

α can then be determined with Eq. (B.26) by

$$\int_{\phi_{\min}}^{\phi_{\max}} \frac{ds^*(\phi)}{d\phi} d\phi = 0 \tag{B.27}$$

$$\Leftrightarrow s^*(\phi_{\max}) - s^*(\phi_{\min}) = 0 \tag{B.28}$$

$$\Leftrightarrow \quad s^*(\phi_{\min}) = s^*(\phi_{\max}) \tag{B.29}$$

$$\Leftrightarrow \quad \frac{s(\phi_{\min})}{\alpha^{\phi_{\min}}} = \frac{s(\phi_{\max})}{\alpha^{\phi_{\max}}} \tag{B.30}$$

$$\Leftrightarrow \quad \alpha = \left(\frac{s(\phi_{\max})}{s(\phi_{\min})}\right)^{\frac{1}{\phi_{\max}-\phi_{\min}}} \tag{B.31}$$

In practice, when only a limited number of spectra at pitches ϕ_d are available, α can be estimated by:

$$\alpha = \left(\frac{s(\phi_D)}{s(\phi_1)}\right)^{\frac{1}{\phi_D-\phi_1}} = \left(\frac{s_D}{s_1}\right)^{\frac{1}{\phi_D-\phi_1}} \tag{B.32}$$

To our best knowledge, the second ambiguity has not been addressed in any of the papers that apply the source-filter model.

Appendix C

Derivations for the database template adaptation

The adaptation of database templates was given by Eq. (4.5) as

$$\mathbf{w}_{\text{data},d} \approx \hat{\mathbf{w}}_{\text{data},d} = \mathbf{w}_{\text{DB},d} \bullet \mathbf{f}. \tag{C.1}$$

The update equation for the filter response $\mathbf{f}$ is derived by applying gradient descent to the β-divergence $C_\beta\left(\mathbf{W}_{\text{data}}, \hat{\mathbf{W}}_{\text{data}}\right)$ between the database spectra and the spectra estimated from the recording. $\mathbf{W}_{\text{data}}$ and $\hat{\mathbf{W}}_{\text{data}}$ denote the matrices containing the templates $\mathbf{w}_{\text{data},d}$ and $\hat{\mathbf{w}}_{\text{data},d}$ for all d. The gradient is given by

$$\frac{\partial C_\beta}{\partial f_k} = \frac{\partial}{\partial f_k} \sum_{k'=0}^{K-1} \sum_{d=0}^{D-1} \frac{w_{\text{data},d,k'}^{\beta}}{\beta(\beta-1)} + \frac{\hat{w}_{\text{data},d,k'}^{\beta}}{\beta} - \frac{w_{\text{data},d,k'} \hat{w}_{\text{data},d,k'}^{\beta-1}}{\beta-1} \tag{C.2}$$

$$= \sum_{d=0}^{D-1} \hat{w}_{\text{data},d,k}^{\beta-1} \frac{\partial \hat{w}_{\text{data},d,k}}{\partial f_k} - w_{\text{data},d,k} \hat{w}_{\text{data},d,k}^{\beta-2} \frac{\partial \hat{w}_{\text{data},d,k}}{\partial f_k} \tag{C.3}$$

$$= \sum_{d=0}^{D-1} \hat{w}_{\text{data},d,k}^{\beta-1} w_{\text{DB},d,k} - w_{\text{data},d,k} \hat{w}_{\text{data},d,k}^{\beta-2} w_{\text{DB},d,k} \tag{C.4}$$

Since $\mathbf{w}_{\text{data},d}$ does not depend on f_k, the first term in Eq. C.2 vanishes in Eq. C.3. Setting the gradient to zero, we obtain:

$$\sum_{d=0}^{D-1} \hat{w}_{\text{data},d,k}^{\beta-1} w_{\text{DB},d,k} = \sum_{d=0}^{D-1} w_{\text{data},d,k} \hat{w}_{\text{data},d,k}^{\beta-2} w_{\text{DB},d,k} \tag{C.5}$$

$$f_k^{\beta-1} \sum_{d=0}^{D-1} \hat{w}_{\text{DB},d,k}^{\beta-1} w_{\text{DB},d,k} = f_k^{\beta-2} \sum_{d=0}^{D-1} w_{\text{data},d,k} \hat{w}_{\text{DB},d,k}^{\beta-2} w_{\text{DB},d,k} \tag{C.6}$$

$$f_k = \frac{\sum_{d=0}^{D-1} w_{\text{data},d,k} \hat{w}_{\text{DB},d,k}^{\beta-1}}{\sum_{d=0}^{D-1} \hat{w}_{\text{DB},d,k}^{\beta}} \tag{C.7}$$

Appendix D

Derivations for the phase-based instrument models

D.1 Time-frequency representation of the model

In Section 6.2, the steady-state part of a harmonic sound was expressed in the time domain by Eq. (6.2) as

$$s(t) = \sum_{p=1}^{P} a_p e^{j[p \cdot \theta(t) + \Delta\varphi_p]}. \tag{D.1}$$

In this section, the time-frequency representation $S(n, k)$ of signal $s(t)$ is derived by calculating its short-time Fourier transform. The STFT is given in its general form by (cf. (6.3))

$$X(n, k) = \sum_{t=-K}^{K-1} x(t + n \cdot m) \cdot h(t) \cdot e^{-j\Omega_k t}, \tag{D.2}$$

where $x(t)$ is the signal under analysis and n and k represent the time frame and frequency index, respectively. $h(t)$ denotes the analysis window of time support $[-K \ldots K-1]$. The distance between consecutive audio frames in samples (hop size) is denoted by m. $\Omega_k = \frac{2\pi k}{2K}$ is the normalised angular frequency of the k-th frequency index.

The STFT of $s(t)$ is hence given by substituting Eq. (D.1) in Eq. (D.2):

$$S(n,k) = \sum_{t=-K}^{K-1} s(t+n\cdot m)\cdot h(t)\cdot e^{-j\Omega_k t} \tag{D.3}$$

$$= \sum_{t=-K}^{K-1}\sum_{p=1}^{P} a_p\cdot e^{j[p\cdot\theta(t+n\cdot m)+\Delta\varphi_p]}\cdot h(t)\cdot e^{-j\Omega_k t} \tag{D.4}$$

$$= \sum_{t=-K}^{K-1}\sum_{p=1}^{P} a_p\cdot e^{j[p\cdot(\omega_1\cdot(t+n\cdot m)+\varphi_1)+\Delta\varphi_p]}\cdot h(t)\cdot e^{-j\Omega_k t} \tag{D.5}$$

$$= \sum_{p=1}^{P} a_p\cdot e^{j[p\cdot(\omega_1\cdot n\cdot m+\varphi_1)]}\cdot e^{j\Delta\varphi_p}\underbrace{\sum_{t=-K}^{K-1} h(t)\cdot e^{j\cdot p\omega_1 t}\cdot e^{-j\Omega_k t}}_{H(\Omega_k-p\omega_1)} \tag{D.6}$$

$$= \sum_{p=1}^{P} a_p\cdot H(\Omega_k-p\omega_1)\cdot e^{j[p\Theta(n)+\Delta\varphi_p]} \tag{D.7}$$

In (D.6) we made use of the frequency shift property of the DFT, which is given by (Oppenheim et al., 1999)

$$h(t)\cdot e^{j\Omega_0 t} \circ\!\!-\!\!\bullet\, H(\Omega-\Omega_0). \tag{D.8}$$

The term $\Theta(n)$ is equivalent to $\theta(n\cdot m)$.

D.2 Parameter estimation

The model parameters are estimated by finding the zeros of the first derivative of the cost function w.r.t. each parameter. The cost function is given by (cf. Eq. (6.10))

$$J = \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} \left| \ln(B(n,k)) - \ln(\hat{B}(n,k)) \right|^2 \tag{D.9}$$

$$= \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} \left| \ln\left[B(n,k)\right] - \ln\left[g(n) \cdot a_{p_k} \cdot H(\Omega_k - p_k\omega_1) \cdot e^{j\left(p_k\Theta(n) + \Delta\varphi_{p_k}\right)} \right] \right|^2 \tag{D.10}$$

$$= \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} \left| \ln\left(\frac{|B(n,k)|}{g(n) \cdot a_{p_k} \cdot H(\Omega_k - p_k\omega_1)} \right) + j\left[\angle B(n,k) - p_k\Theta(n) - \Delta\varphi_{p_k} + 2\pi q(n,k)\right] \right|^2 \tag{D.11}$$

$$= \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} \left[\ln\left(\frac{|B(n,k)|}{g(n) \cdot a_{p_k} H(\Omega_k - p_k\omega_1)} \right) \right]^2 + \left[\angle B(n,k) - p_k\Theta(n) - \Delta\varphi_{p_k} + 2\pi q(n,k)\right]^2, \tag{D.12}$$

where $B(n,k)$ denotes the complex spectrogram and $\hat{B}(n,k) = g(n) \cdot S'(n,k)$ the approximation by the model.

D.2.1 Gains $g(n)$

The derivative of the cost function w.r.t. the gain $g(n)$ is given by

$$\frac{\partial J}{\partial g(n)} = \frac{\partial}{\partial g(n)} \sum_{n'=0}^{N-1} \sum_{k=0}^{K-1} \left[\ln\left(\frac{|B(n',k)|}{g(n') \cdot a_{p_k} H(\Omega_k - p_k\omega_1)} \right) \right]^2 + \left[\angle B(n',k) - p_k\Theta(n') - \Delta\varphi_{p_k} + 2\pi q(n',k)\right]^2 \tag{D.13}$$

$$= \frac{\partial}{\partial g(n)} \sum_{k=0}^{K-1} \left[\ln\left(\frac{|B(n,k)|}{g(n) \cdot a_{p_k} H(\Omega_k - p_k\omega_1)} \right) \right]^2 \tag{D.14}$$

$$= -\frac{2}{g(n)} \sum_{k=0}^{K-1} \ln\left(\frac{|B(n,k)|}{g(n) \cdot a_{p_k} H(\Omega_k - p_k\omega_1)} \right) \tag{D.15}$$

This derivative becomes 0 when either the factor $\frac{2}{g(n)}$ or the sum over k becomes 0. Since the term $\frac{2}{g(n)}$ is nonzero for all $g(n) < \infty$, the derivative in Eq. (D.15) will only vanish when the sum over k becomes 0:

$$0 = \sum_{k=0}^{K-1} \ln\left(\frac{|B(n,k)|}{g(n) \cdot a_{p_k} H(\Omega_k - p_k\omega_1)}\right) \tag{D.16}$$

$$\Rightarrow \ln(g(n)) = \frac{1}{K} \sum_{k=0}^{K-1} \ln\left(\frac{|B(n,k)|}{a_{p_k} H(\Omega_k - p_k\omega_1)}\right) \tag{D.17}$$

$$\Rightarrow g(n) = \exp\left\{\frac{1}{K} \ln\left(\prod_{k=0}^{K-1} \frac{|B(n,k)|}{a_{p_k} H(\Omega_k - p_k\omega_1)}\right)\right\} \tag{D.18}$$

$$= \left(\prod_{k=0}^{K-1} \frac{|B(n,k)|}{a_{p_k} H(\Omega_k - p_k\omega_1)}\right)^{\frac{1}{K}} \tag{D.19}$$

Equation (D.19) calculates the geometric mean of the fraction $\frac{|B(n,k)|}{a_{p_k} H(\Omega_k - p_k\omega_1)}$ over all frequency bins in time frame n.

D.2.2 Partial amplitudes a_p

The derivative for a_p can be obtained in a similar way as for $g(n)$:

$$\frac{\partial J}{\partial a_p} = \frac{\partial}{\partial a_p} \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} \left[\ln\left(\frac{|B(n,k)|}{g(n) \cdot a_{p_k} H(\Omega_k - p_k\omega_1)}\right)\right]^2 + \left[\angle B(n,k) - p_k\Theta(n) - \Delta\varphi_{p_k} + 2\pi q(n,k)\right]^2 \tag{D.20}$$

$$= \frac{\partial}{\partial a_p} \sum_{n=0}^{N-1} \sum_{\{k|p_k=p\}} \left[\ln\left(\frac{|B(n,k)|}{g(n) \cdot a_p H(\Omega_k - p\omega_1)}\right)\right]^2 \tag{D.21}$$

$$= -\frac{2}{a_p} \sum_{n=0}^{N-1} \sum_{\{k|p_k=p\}} \ln\left(\frac{|B(n,k)|}{g(n) \cdot a_p H(\Omega_k - p\omega_1)}\right) \tag{D.22}$$

The term $\frac{2}{a_p}$ is nonzero when $a_p < \infty$, hence the derivative only vanishes when the remaining term becomes 0:

$$0 = \sum_{n=0}^{N-1} \sum_{\{k|p_k=p\}} \ln\left(\frac{|B(n,k)|}{g(n) \cdot a_p H(\Omega_k - p\omega_1)}\right) \tag{D.23}$$

$$\Rightarrow \ln(a_p) = \frac{1}{N \cdot \#\{k|p_k=p\}} \sum_{n=0}^{N-1} \sum_{\{k|p_k=p\}} \ln\left(\frac{|B(n,k)|}{g(n) \cdot H(\Omega_k - p_k\omega_1)}\right) \tag{D.24}$$

$$\Rightarrow a_p = \exp\left\{\frac{1}{N \cdot \#\{k|p_k=p\}} \ln\left(\prod_{n=0}^{N-1} \prod_{\{k|p_k=p\}} \frac{|B(n,k)|}{g(n) \cdot H(\Omega_k - p_k\omega_1)}\right)\right\} \tag{D.25}$$

$$= \left(\prod_{n=0}^{N-1} \prod_{\{k|p_k=p\}} \frac{|B(n,k)|}{g(n) \cdot H(\Omega_k - p_k\omega_1)}\right)^{\frac{1}{N \cdot \#\{k|p_k=p\}}} \tag{D.26}$$

This equation calculates the geometric mean of the fraction $\frac{|B(n,k)|}{g(n) \cdot H(\Omega_k - p_k\omega_1)}$ over all time frames and the relevant frequency bins.

D.2.3 Instantaneous phase of the fundamental $\Theta(n)$

The derivative w.r.t. $\Theta(n)$ is given as follows:

$$\frac{\partial J}{\partial \Theta(n)} = \frac{\partial}{\partial \Theta(n)} \sum_{n'=0}^{N-1} \sum_{k=0}^{K-1} \left[\ln\left(\frac{|B(n',k)|}{g(n') \cdot a_{p_k} H(\Omega_k - p_k\omega_1)}\right)\right]^2 + \left[\angle B(n',k) - p_k\Theta(n') - \Delta\varphi_{p_k} + 2\pi q(n',k)\right]^2 \tag{D.27}$$

$$= \frac{\partial}{\partial \Theta(n)} \sum_{k=0}^{K-1} \left[\angle B(n,k) - p_k\Theta(n) - \Delta\varphi_{p_k} + 2\pi q(n,k)\right]^2 \tag{D.28}$$

$$= -2 \sum_{k=0}^{K-1} \left[\angle B(n,k) - p_k\Theta(n) - \Delta\varphi_{p_k} + 2\pi q(n,k)\right] \cdot p_k \tag{D.29}$$

$$= -2 \sum_{k=0}^{K-1} p_k \cdot \left[\angle B(n,k) - \Delta\varphi_{p_k} + 2\pi q(n,k)\right] - p_k^2\Theta(n) \tag{D.30}$$

Setting the derivative to 0, we obtain

$$\Theta(n) = \frac{\sum_{k=0}^{K-1} p_k \cdot \left[\angle B(n,k) - \Delta\varphi_{p_k} + 2\pi q(n,k)\right]}{\sum_{k=0}^{K-1} p_k^2} \tag{D.31}$$

D.2.4 Relative phase offsets $\Delta\varphi_p$

Differentiating the cost function w.r.t. the relative phase offsets $\Delta\varphi_p$, we obtain

$$\frac{\partial J}{\partial \Delta\varphi_p} = \frac{\partial}{\partial \Delta\varphi_p} \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} \left[\ln\left(\frac{|B(n,k)|}{g(n) \cdot a_{p_k} H(\Omega_k - p_k\omega_1)} \right) \right]^2 + \left[\angle B(n,k) - p_k\Theta(n) - \Delta\varphi_{p_k} + 2\pi q(n,k) \right]^2 \quad \text{(D.32)}$$

$$= \frac{\partial}{\partial \Delta\varphi_p} \sum_{n=0}^{N-1} \sum_{\{k|p_k=p\}} \left[\angle B(n,k) - p\Theta(n) - \Delta\varphi_p + 2\pi q(n,k) \right]^2 \quad \text{(D.33)}$$

$$= -2 \sum_{n=0}^{N-1} \sum_{\{k|p_k=p\}} \left[\angle B(n,k) - p\Theta(n) - \Delta\varphi_p + 2\pi q(n,k) \right] \quad \text{(D.34)}$$

$$= -2 \sum_{n=0}^{N-1} \sum_{\{k|p_k=p\}} \left[\angle B(n,k) - p\Theta(n) + 2\pi q(n,k) \right] + 2N \cdot \#\{k|p_k = p\} \cdot \Delta\varphi_p. \quad \text{(D.35)}$$

This derivative becomes 0 when

$$\Delta\varphi_p = \frac{\sum_{n=0}^{N-1} \sum_{\{k|p_k=p\}} \left[\angle B(n,k) - p\Theta(n) + 2\pi q(n,k) \right]}{N \cdot \#\{k|p_k = p\}}. \quad \text{(D.36)}$$

D.2.5 Phase ambiguity term $q(n,k)$

The derivative of the cost function w.r.t. $q(n,k)$ is given by:

$$\frac{\partial J}{\partial q(n,k)} = \frac{\partial}{\partial q(n,k)} \sum_{n'=0}^{N-1} \sum_{k'=0}^{K-1} \left[\ln\left(\frac{|B(n',k')|}{g(n') \cdot a_{p_{k'}} H(\Omega_{k'} - p_{k'}\omega_1)} \right) \right]^2 + \left[\angle B(n',k') - p_{k'}\Theta(n') - \Delta\varphi_{p_{k'}} + 2\pi q(n',k') \right]^2 \quad \text{(D.37)}$$

$$= \frac{\partial}{\partial q(n,k)} \left(\angle B(n,k) - p\Theta(n) - \Delta\varphi_p + 2\pi q(n,k) \right)^2 \quad \text{(D.38)}$$

$$= 4\pi \cdot \left(\angle B(n,k) - p\Theta(n) - \Delta\varphi_p + 2\pi q(n,k) \right). \quad \text{(D.39)}$$

Setting this equation to 0, and solving for $q(n,k)$ we obtain

$$q(n,k) = -\frac{1}{2\pi} \left(\angle B(n,k) - p\Theta(n) - \Delta\varphi_p \right). \quad \text{(D.40)}$$

Since $q(n,k)$ is supposed to be an integer value, the solution is given by

$$q(n,k) = \left\lfloor -\frac{1}{2\pi} \left(\angle B(n,k) - p\Theta(n) - \Delta\varphi_p \right) \right\rceil , \tag{D.41}$$

where $\lfloor \ldots \rceil$ rounds the argument to the nearest integer.

Bibliography

S. A. Abdallah and M. D. Plumbley. Polyphonic music transcription by non-negative sparse coding of power spectra. In *5th International Conference on Music Information Retrieval*, pages 318–325, Barcelona, Spain, 2004.

ANSI S1.1-1994 (R2004). *Acoustical Terminology*. American National Standards Institute and Acoustical Society of America, 1994.

F. Argenti, P. Nesi, and G. Pantaleo. Automatic transcription of polyphonic music based on the constant-Q bispectral analysis. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(6):1610–1630, 2011.

B. S. Atal and S. L. Hanauer. Speech analysis and synthesis by linear prediction of the speech wave. *Journal of the Acoustical Society of America*, 50(2B): 637–655, 1971.

R. Badeau. Gaussian modeling of mixtures of non-stationary signals in the time-frequency domain (HR-NMF). In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 253–256, New Paltz, USA, October 2011. IEEE.

R. Badeau. High resolution NMF for modeling mixtures of non-stationary signals in the time-frequency domain. Technical Report 2012D004, Télécom ParisTech, Paris, France, July 2012.

D. Barry, B. Lawlor, and E. Coyle. Real-time sound source separation: Azimuth discrimination and resynthesis. In *Audio Engineering Society 117th Convention*, San Francisco, USA, October 2004.

M. Bay, A. F. Ehmann, J. W. Beauchamp, P. Smaragdis, and J. S. Downie. Second fiddle is important too: Pitch tracking individual voices in polyphonic music. In *13th International Society for Music Information Retrieval Conference*, pages 319–324, Porto, Portugal, October 2012.

J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler. A tutorial on onset detection in music signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 13(5):1035–1047, 2005.

J. P. Bello, L. Daudet, and M. B. Sandler. Automatic piano transcription using frequency and time-domain information. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(6):2242–2251, 2006.

E. Benetos and S. Dixon. A temporally-constrained convolutive probabilistic model for pitch detection. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 133–136, New Paltz, USA, 2011a.

E. Benetos and S. Dixon. Joint multi-pitch detection using harmonic envelope estimation for polyphonic music transcription. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1111 –1123, October 2011b.

E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri. Automatic music transcription: challenges and future directions. *Journal of Intelligent Information Systems*, 41(3):407–434, 2013.

E. Benetos. *Automatic transcription of polyphonic music exploiting temporal evolution*. PhD thesis, Queen Mary University of London, 2012.

E. Benetos and S. Dixon. Multiple-instrument polyphonic music transcription using a temporally constrained shift-invariant model. *Journal of the Acoustical Society of America*, 133(3):1727–1741, March 2013.

E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri. Automatic music transcription: Breaking the glass ceiling. In *13th International Society for Music Information Retrieval Conference*, pages 379–384, Porto, Portugal, October 2012.

K. W. Berger. Some factors in the recognition of timbre. *The Journal of the Acoustical Society of America*, 36(10):1888–1891, 1964.

N. Bertin, R. Badeau, and G. Richard. Blind signal decompositions for automatic transcription of polyphonic music: NMF and K-SVD on the benchmark. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Honululu, USA, 2007.

T. Blumensath and M. Davies. Unsupervised learning of sparse and shift-invariant decompositions of polyphonic music. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, 2004.

S. Böck and M. Schedl. Polyphonic piano note transcription with recurrent neural networks. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 121–124, Kyoto, Japan, 2012.

B. P. Bogert, M. J. Healy, and J. W. Tukey. The quefrency alanysis of time series for echoes: Cepstrum, pseudo-autocovariance, cross-cepstrum and saphe cracking. In *Proceedings of the Symposium on Time Series Analysis*, pages 209–243. Wiley, New York, 1963.

Q. D. Bowers. *Encyclopedia of Automatic Musical Instruments*. Vestal Press, 1972.

A. Bregman. *Auditory Scene Analysis: The Perceptual Organization of Sound*. MIT Press, 1994.

N. J. Bryan and G. J. Mysore. Interactive refinement of supervised and semi-supervised sound source separation estimates. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 883–887, Vancouver, Canada, May 2013.

J. J. Burred. *From Sparse Models to Timbre Learning: New Methods for Musical Source Separation*. PhD thesis, Technical University Berlin, 2008.

J. J. Burred and A. Röbel. A segmental spectro-temporal model of musical timbre. In *International Conference on Digital Audio Effects*, Graz, Austria, September 2010.

A. Caclin, S. McAdams, B. K. Smith, and S. Winsberg. Acoustic correlates of timbre space dimensions: A confirmatory study using synthetic tonesa). *Journal of the Acoustical Society of America*, 118(1):471–482, 2005.

M. Caetano and X. Rodet. A source-filter model for musical instrument sound transformation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 137–140, Kyoto, Japan, March 2012.

F. Cañadas Quesada, P. Vera-Candeas, N. Ruiz-Reyes, R. Mata-Campos, and J. Carabias-Orti. Note-event detection in polyphonic musical signals based on harmonic matching pursuit and spectral smoothness. *Journal of New Music Research*, 37(3):167–183, 2008.

F. Cañadas Quesada, N. Ruiz Reyes, P. Vera Candeas, J. Carabias, and S. Maldonado. A multiple-f0 estimation approach based on gaussian spectral modelling for polyphonic music transcription. *Journal of New Music Research*, 39(1): 93–107, 2010.

C. Chafe, D. Jaffe, K. Kashima, B. Mont-Reynaud, and J. O. Smith. Techniques for note identification in polyphonic music. In *International Computer Music Conference*, Vancouver, Canada, 1985.

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal of Scientific Computing*, 20(1):33–61, 1998.

A. de Cheveigné. Multiple f0 estimation. In D. Wang and G. J. Brown, editors, *Computational Auditory Scene Analysis: Principles, Algorithms and Applications*, pages 45–79. IEEE Press/Wiley, 2006.

A. de Cheveigné and H. Kawahara. Multiple period estimation and pitch perception model. *Speech Communication*, 27(3-4):175–185, 1999.

A. de Cheveigné and H. Kawahara. YIN, a fundamental frequency estimator for speech and music. *Journal of the Acoustical Society of America*, 111(4): 1917–1930, 2002.

A. Cichocki, R. Zdunek, and S.-i. Amari. Csiszar's divergences for non-negative matrix factorization: Family of new algorithms. In *6th International Conference on Independent Component Analysis and Blind Source Separation*, pages 32–39, Charleston, USA, 2006.

A. Cont. Realtime multiple pitch observation using sparse non-negative constraints. In *7th International Conference on Music Information Retrieval*, Victoria, Canada, 2006.

A. Cont, S. Dubnov, and D. Wessel. Realtime multiple-pitch and multiple-instrument recognition for music signals using sparse non-negative constraints. In *International Conference on Digital Audio Effects*, pages 85–92, Bordeaux, France, September 2007.

G. Costantini, R. Perfetti, and M. Todisco. Event based transcription system for polyphonic piano music. *Signal Processing*, 89(9):1798–1811, 2009.

M. Davy, S. Godsill, and J. Idier. Bayesian analysis of polyphonic western tonal music. *Journal of the Acoustical Society of America*, 119:2498, 2006.

A. Dessein, A. Cont, and G. Lemaitre. Real-time polyphonic music transcription with non-negative matrix factorization and beta-divergence. In *11th International Society for Music Information Retrieval Conference*, Utrecht, The Netherlands, August 2010.

C. Dittmar and J. Abeßer. Automatic music transcription with user interaction. In *Deutsche Jahrestagung für Akustik*, pages 567–568, Dresden, Germany, 2008.

S. Dixon. An interactive beat tracking and visualisation system. In *International Computer Music Conference*, Havana, Cuba, 2001.

K. Dressler. Pitch estimation by the pair-wise evaluation of spectral peaks. In *AES 42nd Conference on Semantic Audio*, Ilmenau, Germany, July 2011.

K. Dressler. Multiple fundamental frequency extraction for MIREX 2012. In *Music Information Retrieval Evaluation eXchange (MIREX)*, 2012.

Z. Duan, B. Pardo, and C. Zhang. Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(8):2121–2133, 2010.

Z. Duan, J. Han, and B. Pardo. Harmonically informed multi-pitch tracking. In *10th International Society for Music Information Retrieval Conference*, pages 333–338, Kobe, Japan, October 2009.

H. Dudley. Remaking speech. *Journal of the Acoustical Society of America*, 11: 169, 1939.

J. Durrieu and J. Thiran. Musical audio source separation based on user-selected f0 track. In *10th International Conference on Latent Variable Analysis and Signal Separation*, Tel-Aviv, Israel, March 2012.

D. P. W. Ellis. *Prediction-driven Computational Auditory Scene Analysis*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, June 1996.

V. Emiya, R. Badeau, and B. David. Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6):1643–1654, 2010.

G. T. Fechner. *Elemente der Psychophysik*. Breitkopf & Härtel, Leipzig, Germany, 1860.

Federal Standard 1037C. *Telecommunications: Glossary of Telecommunication Terms*. General Services Administration, 1996.

C. Févotte, N. Bertin, and J. Durrieu. Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis. *Neural Computation*, 21(3):793–830, 2009.

D. FitzGerald, M. Cranitch, and E. Coyle. Shifted non-negative matrix factorisation for sound source separation. In *IEEE 13th Workshop on Statistical Signal Processing*, pages 1132–1137, Bordeaux, France, 2005.

N. H. Fletcher. Mode locking in nonlinearly excited inharmonic musical oscillators. *Journal of the Acoustical Society of America*, 64:1566, 1978.

N. H. Fletcher and D. Rossing. *The Physics of Musical Instruments*. Springer, New York, 2nd edition, 1991.

J. Fritsch. High quality musical audio source separation. Master's thesis, UPMC/IRCAM/Télécom ParisTech, 2012.

B. Fuentes, R. Badeau, and G. Richard. Blind harmonic adaptive decomposition applied to supervised source separation. In *European Signal Processing Conference*, pages 2654—2658, Bucharest, Romania, 2012.

B. Fuentes, R. Badeau, and G. Richard. Adaptive harmonic time-frequency decomposition of audio using shift-invariant PLCA. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 401–404, Prague, Czech Republic, May 2011.

D. Godsmark and G. Brown. A blackboard architecture for computational auditory scene analysis. *Speech Communication*, 27(3-4):351–366, 1999.

C. L. Gonzo. An analysis of factors related to choral teachers' ability to detect pitch errors while reading the score. *Journal of Research in Music Education*, pages 259–271, 1971.

T. Goolsby. Computer applications to eye movement research in music reading. *Psychomusicology: A Journal of Research in Music Cognition*, 8(2):111–126, 1989.

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.

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

J. Grey. Multidimensional perceptual scaling of musical timbres. *Journal of the Acoustical Society of America*, 61(5):1270–1277, 1977.

D. Griffin and J. Lim. Signal reconstruction from short-time Fourier transform magnitude. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 31(4):986–998, August 1983.

G. Grindlay and D. P. W. Ellis. Multi-voice polyphonic music transcription using eigeninstruments. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, USA, 2009.

G. Grindlay and D. P. Ellis. Transcribing multi-instrument polyphonic music with hierarchical eigeninstruments. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1159–1169, 2011.

P. Grosche, B. Schuller, M. Müller, and G. Rigoll. Automatic transcription of recorded music. *Acta Acoustica united with Acoustica*, 98(2):199–215, 2012.

H. Hahn, A. Röbel, J. Burred, and S. Weinzierl. Source-filter model for quasi-harmonic instruments. In *International Conference on Digital Audio Effects*, Graz, Austria, 2010.

S. W. Hainsworth. *Techniques for the Automated Analysis of Musical Audio*. PhD thesis, University of Cambridge, UK, September 2004.

H. H. Hall. Sound analysis. *Journal of the Acoustical Society of America*, 8(4): 257–262, 1937.

L. A. Hansen. A study of score reading ability of musicians. *Journal of Research in Music Education*, 9(2):147–156, 1961.

W. M. Hartmann. Pitch, periodicity, and auditory organization. *Journal of the Acoustical Society of America*, 100:3491, 1996.

T. Heittola, A. Klapuri, and T. Virtanen. Musical instrument recognition in polyphonic audio using source-filter model for sound separation. In *10th International Society for Music Information Retrieval Conference*, pages 327–332, Kobe, Japan, 2009.

R. Hennequin, R. Badeau, and B. David. Time-dependent parametric and harmonic templates in non-negative matrix factorization. In *International Conference on Digital Audio Effects*, pages 246—253, Graz, Austria, September 2010.

R. Hennequin, R. Badeau, and B. David. NMF with time-frequency activations to model nonstationary audio events. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):744–753, May 2011.

M. Hoffman, D. M. Blei, and P. R. Cook. Bayesian nonparametric matrix factorization for recorded music. In *International Conference on Machine Learning*, pages 439–446, Haifa, Israel, 2010.

T. Hofmann. Probabilistic latent semantic indexing. In *22nd International Conference on Research and Development in Information Retrieval (ACM SIGIR)*, pages 50–57, Berkeley, USA, 1999. ACM.

P. Iverson and C. L. Krumhansl. Isolating the dynamic attributes of musical timbre. *Journal of the Acoustical Society of America*, 94(5):2595–2603, 1993.

X. Jaureguiberry, P. Leveau, S. Maller, and J. J. Burred. Adaptation of source-specific dictionaries in non-negative matrix factorization for source separation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 5–8, Prague, Czech Republic, 2011.

H. Kameoka, T. Nishimoto, and S. Sagayama. A multipitch analyzer based on harmonic temporal structured clustering. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(3):982–994, 2007.

H. Kameoka, N. Ono, K. Kashino, and S. Sagayama. Complex NMF: A new sparse representation for acoustic signals. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3437–3440, Taipei, Taiwan, 2009.

K. Kashino and H. Tanaka. A sound source separation system with the ability of automatic tone modeling. In *International Computer Music Conference*, pages 248–248, Tokyo, Japan, 1993.

K. Kashino, K. Nakadai, T. Kinoshita, and H. Tanaka. Organization of hierarchical perceptual sounds: Music scene analysis with autonomous processing modules and a quantitative information integration mechanism. In *International Joint Conference on Artificial Intelligence*, pages 158–164, Montreal, Quebec, 1995.

R. A. Kendall. The role of acoustic signal partitions in listener categorization of musical phrases. *Music perception*, pages 185–213, 1986.

R. A. Kendall and E. C. Carterette. Identification and blend of timbres as a basis for orchestration. *Contemporary Music Review*, 9(1–2):51–67, 1993.

H. Kirchhoff, S. Dixon, and A. Klapuri. Shift-variant non-negative matrix deconvolution for music transcription. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 125–128, Kyoto, Japan, March 2012a.

H. Kirchhoff, S. Dixon, and A. Klapuri. Derivation of update equations for shift-variant non-negative matrix deconvolution (svNMD). Technical Report C4DM-TR-01-12, Queen Mary University of London, 2012b. `http://www.eecs.qmul.ac.uk/~holger/C4DM-TR-01-12`.

H. Kirchhoff, S. Dixon, and A. Klapuri. Multi-template shift-variant non-negative matrix deconvolution for semi-automatic music transcription. In

13th International Society for Music Information Retrieval Conference, pages 415–420, Porto, Portugal, October 2012c.

H. Kirchhoff, S. Dixon, and A. Klapuri. Derivation of update equations for multiple-template shift-variant non-negative matrix deconvolution based on β-divergence. Technical Report C4DM-TR-06-12, Queen Mary University of London, 2012d. `http://www.eecs.qmul.ac.uk/~holger/C4DM-TR-06-12`.

H. Kirchhoff, S. Dixon, and A. Klapuri. Derivation of update equations for a source-filter model based on beta-divergence. Technical Report C4DM-TR-10-12, Queen Mary University of London, 2012e. `http://www.eecs.qmul.ac.uk/~holger/C4DM-TR-10-12`.

H. Kirchhoff, S. Dixon, and A. Klapuri. Cross-recording adaptation of musical instrument spectra. Technical Report C4DM-TR-11-12, Queen Mary University of London, 2012f. `http://www.eecs.qmul.ac.uk/~holger/C4DM-TR-11-12`.

H. Kirchhoff, S. Dixon, and A. Klapuri. Missing template estimation for user-assisted music transcription. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 26–30, Vancouver, Canada, May 2013a.

H. Kirchhoff, S. Dixon, and A. Klapuri. Multiple instrument tracking based on reconstruction error, pitch continuity and instrument activity. In *10th International Symposium on Computer Music Multidisciplinary Research*, Marseille, France, October 2013b.

H. Kirchhoff, R. Badeau, and S. Dixon. Towards complex matrix decomposition of spectrograms based on the relative phase offsets of harmonic sounds. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Florence, Italy, 2014. submitted.

A. Klapuri. Automatic transcription of music. Master's thesis, Tampere University of Technology, 1998.

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

A. Klapuri. Analysis of musical instrument sounds by source-filter-decay model. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Honululu, USA, 2007.

A. Klapuri. Multiple fundamental frequency estimation based on harmonicity and spectral smoothness. *IEEE Transactions on Speech and Audio Processing*, 11(6):804–815, 2003.

A. Klapuri. *Signal Processing Methods for the Automatic Transcription of Music*. PhD thesis, Tampere University of Technology, 2004a.

A. Klapuri and M. Davy, editors. *Signal Processing Methods for Music Transcription*. Springer, 2006.

A. Klapuri. Automatic music transcription as we know it today. *Journal of New Music Research*, 33(3):269–282, 2004b.

A. Klapuri. A perceptually motivated multiple-f0 estimation method. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 291 – 294, New Paltz, USA, 2005.

C.-T. Lee, Y.-H. Yang, and H. Chen. Automatic transcription of piano music by sparse representation of magnitude spectra. In *IEEE International Conference on Multimedia and Expo*, pages 1–6, Barcelona, Spain, 2011.

D. Lee and H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

D. Lee and H. Seung. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*, 13, 2001.

P. Leveau, E. Vincent, G. Richard, and L. Daudet. Instrument-specific harmonic atoms for mid-level music representation. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(1):116–128, 2008.

R. C. Maher. Evaluation of a method for separating digitized duet signals. *Journal of the Audio Engineering Society*, 38(12):956–979, December 1990.

S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.

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

K. D. Martin. A blackboard system for automatic transcription of simple polyphonic music. Technical Report 385, MIT Media Laboratory Perceptual Computing Section, 1996a.

K. D. Martin. Automatic transcription of simple polyphonic music - robust front end processing. Technical Report 399, MIT Media Laboratory Perceptual Computing Section, 1996b.

K. D. Martin. *Sound-source recognition: a theory and computational model*. PhD thesis, Massachusetts Institute of Technology, MA, USA, June 1999.

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

S. McAdams, S. Winsberg, S. Donnadieu, G. De Soete, and J. Krimphoff. Perceptual scaling of synthesized musical timbres: Common dimensions, specificities, and latent subject classes. *Psychological Research*, 58(3):177–192, 1995.

R. Meddis and M. Hewitt. Virtual pitch and phase sensitivity of a computer model of the auditory periphery. I: Pitch identification. *Journal of the Acoustical Society of America*, 89(6):2866–2882, 1991.

R. Meddis and L. O'Mard. A unitary model of pitch perception. *Journal of the Acoustical Society of America*, 102, 1997.

E. Meyer and G. Buchmann. Die Klangspektren der Musikinstrumente. *Sitzungsberichte der Preußischen Akademie der Wissenschaften*, XXXII. Sitzung der Physikalisch-Mathematischen Klasse, 1931.

J. R. Miller and E. C. Carterette. Perceptual space for musical structures. *Journal of the Acoustical Society of America*, 58(3):711–720, 1975.

B. C. J. Moore, editor. *Hearing. Handbook of Perception and Cognition*. Academic Press, San Diego, California, 2nd edition, 1995.

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

J. A. Moorer. *On the segmentation and analysis of continuous musical sound by digital computer*. PhD thesis, Center for Computer Research in Music and Acoustics, Stanford University, 1975.

E. Moulines and J. Laroche. Non-parametric techniques for pitch-scale and time-scale modification of speech. *Speech communication*, 16(2):175–205, 1995.

G. J. Mysore and P. Smaragdis. Relative pitch estimation of multiple instruments. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 313–316, Taipei, Taiwan, April 2009. IEEE.

G. J. Mysore, P. Smaragdis, and B. Raj. Non-negative hidden Markov modeling of audio with application to source separation. In *9th International Conference on Latent Variable Analysis and Signal Separation*, pages 140–148, St. Malo, France, 2010.

J. Nam, J. Ngiam, H. Lee, and M. Slaney. A classification-based polyphonic piano transcription approach using learned feature representations. In *12th International Society for Music Information Retrieval Conference*, pages 175–180, Miami, USA, 2011.

B. Niedermayer. Non-negative matrix division for the automatic transcription of polyphonic music. In *9th International Conference on Music Information Retrieval*, pages 544–549, 2008.

P. O'Grady and B. Pearlmutter. Convolutive non-negative matrix factorisation with sparseness constraint. In *IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*, 2006.

K. O'Hanlon, H. Nagano, and M. D. Plumbley. Structured sparsity for automatic music transcription. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 441–444, Kyoto, Japan, 2012.

B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision Research*, 37(23):3311–3325, 1997.

A. V. Oppenheim, R. W. Schafer, and J. R. Buck. *Discrete-Time Signal Processing*. Prentice Hall, 2nd edition, 1999.

A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot. One microphone singing voice separation using source-adapted models. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 90–93, New Paltz, USA, 2005.

A. Ozerov, E. Vincent, and F. Bimbot. A general flexible framework for the handling of prior information in audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(4):1118–1133, 2012.

P. Paatero and U. Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.

R. M. Parry and I. Essa. Phase-aware non-negative spectrogram factorization. In *7th International Conference on Independent Component Analysis and Signal Separation*, pages 536–543, London, UK, 2007. Springer.

A. Pertusa and J. Iñesta. Multiple fundamental frequency estimation using gaussian smoothness. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 105–108, Las Vegas, USA, 2008.

A. Pertusa and J. M. Iñesta. Polyphonic monotimbral music transcription using dynamic networks. *Pattern Recognition Letters*, 26(12):1809–1818, 2005.

G. E. Poliner and D. P. Ellis. A discriminative model for polyphonic piano transcription. *EURASIP Journal on Advances in Signal Processing*, 2007a.

G. E. Poliner and D. P. Ellis. Improving generalization for classification-based polyphonic piano transcription. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 86–89, New Paltz, USA, October 2007b.

L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

S. Raczyński, N. Ono, and S. Sagayama. Multipitch analysis with harmonic nonnegative matrix approximation. In *8th International Conference on Music Information Retrieval*, pages 381–386, Vienna, Austria, 2007.

S. A. Raczynski, E. Vincent, F. Bimbot, and S. Sagayama. Multiple pitch transcription using DBN-based musicological models. In *11th International Society for Music Information Retrieval Conference*, pages 363–368, Utrecht, The Netherlands, 2010.

S. A. Raczynski, E. Vincent, and S. Sagayama. Dynamic Bayesian networks for symbolic polyphonic pitch modeling. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(9):1830–1840, 2013.

J.-C. Risset and D. L. Wessel. Exploration of timbre by analysis and synthesis. In D. Deutsch, editor, *The Psychology of Music*, pages 113–169. Academic Press, 1999.

A. Röbel. Between physics and perception: Signal models for high level audio processing. In *International Conference on Digital Audio Effects*, Graz, Austria, September 2010.

M. Ryynänen and A. Klapuri. Polyphonic music transcription using note event modeling. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 319–322, New Paltz, USA, 2005.

S. Saito, H. Kameoka, K. Takahashi, T. Nishimoto, and S. Sagayama. Specmurt analysis of polyphonic music signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(3):639–650, 2008.

G. J. Sandell and W. L. Martens. Perceptual evaluation of principal-component-based synthesis of musical timbres. *Journal of the Audio Engineering Society*, 43(12):1013–1028, 1995.

D. Sarason. *Complex Function Theory*. American Mathematical Society, 2nd edition, 2007.

M. Schmidt and M. Mørup. Nonnegative matrix factor 2-D deconvolution for blind single channel source separation. In *6th International Conference on Independent Component Analysis and Blind Source Separation*, pages 700–707, Chareston, SC, USA, March 2006a. Springer.

M. Schmidt and M. Mørup. Sparse non-negative matrix factor 2-D deconvolution for automatic transcription of polyphonic music. In *International Conference on Independent Component Analysis and Blind Source Separation*, Charleston, USA, 2006b.

C. Schörkhuber and A. Klapuri. Constant-q transform toolbox for music processing. In *Sound and Music Computing Conference*, Barcelona, Spain, 2010.

D. Schwarz. Spectral envelopes in sound analysis and synthesis. Master's thesis, Universität Stuttgart, 1998.

M. Shashanka. *Latent variable framework for modeling and separating single-channel acoustic sources*. PhD thesis, Boston University, 2008.

J. A. Sloboda. Experimental studies of music reading: A review. *Music Perception: An Interdisciplinary Journal*, 2(2):222–236, 1984.

P. Smaragdis. Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs. In *5th International Conference on Independent Component Analysis and Blind Signal Separation*, pages 494–499, Granada, Spain, September 2004.

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

P. Smaragdis and G. J. Mysore. Separation by humming: User-guided sound extraction from monophonic mixtures. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, USA, October 2009.

P. Smaragdis and B. Raj. Shift-invariant probabilistic latent component analysis. Technical Report TR2007-009, Mitsubishi Research Laboratories, 2007.

P. Smaragdis, B. Raj, and M. Shashanka. A probabilistic latent variable model for acoustic modeling. In *Neural Information Processing Systems*, volume 148, Vancouver, Canada, 2006.

A. D. Sterian. *Model-Based Segmentation of Time-Frequency Images for Musical Transcription*. PhD thesis, University of Michigan, 1999.

W. Strong and M. Clark. Synthesis of wind-instrument tones. *Journal of the Acoustical Society of America*, 41(1):39–52, 1967.

C. Stumpf. *Tonpsychologie*, volume 2. S. Hirzel Verlag, 1890.

C. Stumpf. *Die Sprachlaute. Experimentell-phonetische Untersuchungen nebst einem Anhang über Instrumentalklänge*. Springer, 1926.

T. Tolonen and M. Karjalainen. A computationally efficient multipitch analysis model. *IEEE Transactions on Audio, Speech, and Language Processing*, 8(6): 708–716, 2000.

V. Välimäki, J. Pakarinen, C. Erkut, and M. Karjalainen. Discrete-time modelling of musical instruments. *Reports on progress in physics*, 69, 2006.

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

E. Vincent, N. Bertin, and R. Badeau. Adaptive harmonic spectral decomposition for multiple pitch estimation. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3):528–537, 2010.

E. Vincent and X. Rodet. Music transcription with ISA and HMM. In *5th International Conference on Independent Component Analysis and Blind Signal Separation*, pages 1197–1204. Springer, 2004.

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, March 2007.

T. Virtanen and A. Klapuri. Analysis of polyphonic audio using source-filter model and non-negative matrix factorization. In *Advances in Models for Acoustic Processing, Neural Information Processing Systems Workshop*, 2006.

T. Virtanen, A. T. Cemgil, and S. Godsill. Bayesian extensions to non-negative matrix factorisation for audio signal modelling. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 1825–1828, Las Vegas, USA, 2008.

A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, 1967.

H. von Helmholtz. *Die Lehre von den Tonempfindungen als physiologische Grundlage für die Theorie der Musik*. Vieweg & Sohn, 1870.

H. von Helmholtz. *On the sensations of tone as a physiological basis for the theory of music*. (A.J. Ellis, Trans.). Dover, New York, 1954.

L. Wedin and G. Goude. Dimension analysis of the perception of instrumental timbre. *Scandinavian Journal of Psychology*, 13(1):228–240, 1972.

C. Yeh, A. Röbel, and X. Rodet. Multiple fundamental frequency estimation and polyphony inference of polyphonic music signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6):1116–1126, 2010.

C. Yeh. *Multiple Fundamental Frequency Estimation of Polyphonic Recordings*. PhD thesis, Université Paris VI - Pierre et Marie Curie, 2008.

K. Yoshii and M. Goto. A nonparametric bayesian multipitch analyzer based on infinite latent harmonic allocation. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(3):717–730, 2012.

R. Zhou. *Feature extraction of musical content for automatic music transcription*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2006.

E. Zwicker and H. Fastl. *Psychoacoustics*, volume 22. Springer, 1999.