Source Separation and Restoration of Drum Sounds in Music Recordings

Quellentrennung und Restauration von Schlagzeugklängen in Musikaufnahmen

Der Technischen Fakultät

der Friedrich-Alexander-Universität Erlangen-Nürnberg

zur

Erlangung des Doktorgrades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)

vorgelegt von

Christian Dittmar

aus

Jena

Als Dissertation genehmigt
von der Technischen Fakultät
der Friedrich-Alexander-Universität Erlangen-Nürnberg

Tag der mündlichen Prüfung:	8. Mai 2018
Vorsitzender des Promotionsorgans:	Prof. Dr.-Ing. Reinhard Lerch
1. Gutachter:	Prof. Dr. Meinard Müller
2. Gutachter:	Dr. Kazuyoshi Yoshii

Abstract

In the fields of signal processing and music information retrieval (MIR), the task of decomposing a music recording into musically meaningful sound sources is referred to as source separation. As an example, a funk instrumental may consist of drums, bass, and saxophone playing together. Given this mixture, the goal is to recover the individual instruments' sounds—as if they had been recorded in isolation.

In this thesis, we focus on separating and restoring drum sounds from music recordings. Drums typically emphasize and shape the rhythm, and often define the musical style. In contrast to melodic instruments, drums mainly produce percussive and inharmonic sounds that may overlap considerably in time and frequency. The challenge is to extract perceptually convincing source signals without introducing audible artifacts. We systematically approach this research problem in three parts of the thesis, each with a different perspective.

The first part is about automatic drum transcription (ADT), i. e., the detection and classification of drum sound events in music recordings. Based on a literature review, we give a comprehensive account of prior work on this topic. We then select two families of state-of-the-art ADT algorithms based on Non-Negative Matrix Factorization (NMF) and Recurrent Neural Networks (RNN), and compare their performance in a controlled experimental setting.

In the second part of this thesis, we focus on drum sound separation. Aiming for a better understanding of the capabilities and limitations of NMF, we investigate how to embed side information and how to exploit drum-specific properties. As one main contribution, we introduce suitable constraints that help steer iterative NMF methods towards a meaningful solution. Furthermore, to improve the perceptual quality of the separated drum sounds, we propose dictionary-based restoration schemes for repairing cross-talk artifacts. Finally, we investigate signal reconstruction methods and develop a transient restoration technique that is suited to sharpen the attack region of drum sounds.

In the third part, we consider an application of automated methods to interdisciplinary microtiming research, particularly swing ratio estimation in jazz performances. Throughout this thesis, we apply our decomposition and restoration approaches to music analysis and editing tasks. By exploring novel algorithmic approaches for drum sound separation within concrete application scenarios, this thesis contributes to fundamental research of theoretical and practical relevance.

Zusammenfassung

Der Begriff der Quellentrennung steht in den Bereichen der Signalverarbeitung und des Music Information Retrieval (MIR) für die Zerlegung von Musiksignalen in ihre Klangbestandteile. Wenn beispielsweise in der Aufnahme einer Funk-Band Schlagzeug, Bass und Saxophon gemeinsam spielen, wäre das Ziel, die einzelnen Instrumentenklänge so aus der Mischung zu rekonstruieren, als ob diese einzeln aufgenommen wurden.
Die vorliegende Arbeit befasst sich mit der Quellentrennung und Restauration von Schlagzeugklängen. In vielen Musikstilen wird der Rhythmus durch Schlaginstrumente vorgegeben. Im Gegensatz zu Melodieinstrumenten erzeugen diese vorwiegend perkussive und inharmonische Klänge, die zeitlich und spektral große Überlappungen aufweisen können. Die Herausforderung besteht darin, perzeptuell hochwertige Teilsignale ohne hörbare Artefakte zu extrahieren. In der vorliegenden Arbeit nähern wir uns dieser Aufgabe in drei Teilen, in denen wir systematisch verschiedende Aspekte bearbeiten.
Der erste Teil befasst sich mit der automatisierten Transkription von Schlagzeuginstrumenten in Musikaufnahmen (ADT). Auf Basis einer umfangreichen Literaturrecherche geben wir zunächst einen kompakten Überblick über den Forschungsstand zu diesem Thema. Anschließend vergleichen wir in einem kontrollierten Experiment die Leistungsfähigkeit von aktuellen ADT-Ansätzen die auf Nicht-Negativer Matrixfaktorisierung (NMF) und rekurrenten neuronalen Netzen (RNN) basieren.
Im zweiten Teil konzentrieren wir uns auf die Quellentrennung von Schlagzeugklängen. Als ersten Beitrag zeigen wir auf, wie das Wissen um Schlagzeugeigenschaften genutzt werden kann, um iterative NMF-Algorithmen in Richtung einer musikalisch sinnvollen Lösung zu beeinflußen. Weiter schlagen wir Restaurationsmethoden zur Reduzierung von Übersprechen in den extrahierten Klangkomponenten vor. Schließlich untersuchen wir Vefahren zur Signalrekonstruktion und entwickeln einen Ansatz für die Restauration des transienten Einschwingverhaltens von Schlaginstrumenten.
Im dritten Teil leisten wir einen interdisziplinären Beitrag zur computergestützten Erforschung von Microtiming im Jazz, mit besonderem Augenmerk auf Swing Ratio-Schätzung. Durch die Erforschung neuartiger Algorithmen in konkreten Anwendungsszenarion liefert diese Arbeit grundlegende Forschungsbeiträge von sowohl theoretischer als auch praktischer Relevanz.

Contents

Chapter 1

Introduction

In our modern world, music is a ubiquitous part of daily life. Via mobile devices and the internet, we can access millions of music recordings almost instantly. Although it often serves as an inconspicuous sound backdrop for TV shows and advertisements, music can also have a very intimate meaning to listeners. When listening consciously to music, humans have the ability to focus on certain instruments or voices, despite the superposition of acoustic source signals from the individual instruments—in a form that seemingly could not be disentangled. According to Bregman [20], the human auditory system organizes the audio signal into perceptual streams by first decomposing it into elementary units before selectively grouping these units into the source streams.

If computers could do the same, we would be able to interact with digital music recordings in novel ways. Instead of being a monolithic time-series of discretized and quantized acoustic waves, digital music recordings could be treated as a composition of individual parts, each of which could be auditioned, analyzed, manipulated, and re-used individually. A driving force behind this scenario is my own interest in digital music production. Technologies such as Stems[1] show that there is indeed a commercial demand for such possibilities. As laid out in [109], technological advancements such as the availability of affordable digital samplers to the wider public have had a strong influence on music creation practices and gave rise to entirely new music genres. Affordable (or even free) music production software has put the capabilities of music recording studios into the hands of everyday users. Especially in electronic music genres, re-using recorded music material has been a major source of inspiration. For a long time, creative use of music samples has been constrained to pitch shifting, time stretching, slicing, and resequencing. Over the last few years, innovative music production software such as Melodyne[2] or Regroover[3] have proven that music production can be fundamentally different to conventional paradigms when it

[1] `https://www.native-instruments.com/en/specials/stems/`, last accessed June 14, 2018
[2] `http://www.celemony.com/en/service1/about-celemony/technologies`, last accessed June 14, 2018
[3] `http://regroover.com/`, last accessed June 14, 2018

is possible to decompose music signals into their constituent note events. We can expect to see further technological advances in the near future, and this thesis is intended to contribute to this overall development on the scientific side.

In the broader research areas of audio signal processing and music information retrieval (MIR), the task of (semi-)automatically decomposing a music recording into musically meaningful sound sources is commonly referred to as source separation. As mentioned before, music recordings can be thought of as structured superpositions (or mixtures) of different sound sources. For example, an instrumental recording of a funk band may consist of drums, bass, and saxophone. Given this mixture, the goal of source separation is to derive source signals that correspond to the individual instruments, or even single note events played on these instruments. Ideally, the source signals recovered from the mixture would sound as if they had been recorded in isolation. With a simple thought experiment, one can easily imagine why music source separation poses a very challenging task. Assume we are given an integer number that is the sum of two unknown integers. The task is to recover the summands by staring insistently at the sum. Clearly, the possible solutions are endless if no additional information is given. A tiny bit of side information could be that both numbers are non-negative and non-zero. Immediately, we would see that the summands must be smaller than the sum, significantly reducing the set of possible solutions.

Transferring this analogy to the human auditory system, valuable side information lies in auditory cues (e. g., harmonicity, temporal smoothness, common frequency modulation, etc.). Consequently, many source separation methods exploit specific spectral and temporal properties of the target source signals. For example, the singing voice or lead instrument is often characterized by its spectral dominance in relation to the accompanying instruments. Another example could be the bass, which usually plays the notes with the lowest fundamental frequencies occurring in the mixture. Moreover, certain instruments may be identifiable by characteristic energy distributions in their harmonic series. Although these assumptions are often beneficial for specific tasks, general music source separation is a problem far from being solved. Until recently, research has concentrated mainly on the extraction of pitched instruments, such as the human singing voice, and only few authors have approached the source separation of drum sound components.

In this thesis, we aim to close this gap by focusing on drum kits and the most common drum instruments such as kick drum, snare drum, hi-hat, and ride cymbals. Due to the specific nature of drum and percussion instruments, only some of the afore-mentioned auditory cues are useful (e. g., the notion of overtone series is not applicable to many drum instruments). To make the task feasible, we additionally exploit the availability of prior musical knowledge. For piano music, it has been shown that score information is a very helpful cue [67] in order to achieve perceptually convincing signal decompositions. Recognizing the similarities in temporal structure of drum sounds and piano sounds, we transfer this principle to our scenario. However, as we will show, automatically transcribing the drum part in music mixtures with sufficient accuracy is an open research problem. Therefore, at certain parts of this thesis,

we will assume a perfect transcription was provided by an oracle for guiding the source separation.

1.1 Structure

This thesis is structured in three main parts, covering nine chapters. The individual parts represent interconnected research topics. Methods and concepts developed in the first part also lay important foundations for the two subsequent parts. The first part is about automatic drum transcription (ADT), i. e., the automatic detection and classification of drum sound events in music recordings. In the opening Chapter 2, we first discuss the particularities and challenges of processing recordings of drum instruments and then provide an in-depth overview of ADT publications over the last 15 years. Many fundamental techniques are discussed that are important throughout the entire thesis. Then, in Chapter 3, we will focus on two families of ADT methods, which we subsume under the umbrella term *activation-based* methods. On the one hand, these techniques are based on different flavours of Non-Negative Matrix Factorization (NMF). On the other hand, different architectures of Recurrent Neural Networks (RNN) are prominently used. Therefore, we will describe the setup and main results of a well-defined performance comparison between both families of algorithms.
In the second part of the thesis, we will treat diverse aspects of music source separation with a strong focus on drum sound separation. We start in Chapter 4 with score-informed decomposition of drum-only recordings. To this end, we assess the suitability of Non-Negative Matrix Factor Deconvolution (NMFD). We circumvent the problem of potentially imperfect drum sound detection by using oracle score information. Our contributions in this chapter are twofold. First, we aim to foster a better understanding of the capabilities and limitations of NMFD, and the ways to embed side information into the decomposition. Second, we propose two dictionary-based restoration methods and evaluate the merits of using them for improving perceptual separation quality. We refer to this concept of first decomposing and afterward repairing time-frequency representations as *separate and restore*.
We further extend this separate-and-restore concept in Chapter 5. In this context, we aim at repairing cross-talk artifacts that can occur in score-informed decompositions of piano recordings. First, we recapitulate a framework for decomposing a music recording with respect to note events as specified in a given musical score. The intermediate results are then used to transfer the spectral envelope of imperfectly separated note events to perfectly isolated piano notes from a dictionary.
After that, we investigate a classic iterative phase-reconstruction method in Chapter 6. We show that simply using the mixture phase is a reasonable choice for drum sound separation. To further improve the quality of the transient signal portions, we propose a novel modification enforcing time-domain constraints to intermediate signal reconstructions during the phase-reconstruction

iterations. We refer to the resulting algorithm as *transient restoration*.
With Chapter 7, we turn our attention towards generalized Wiener filtering. In the first part of this chapter, we explore—through a series of source separation experiments with oracle component signals—how the optimal choice of the magnitude exponent is influenced by signal types, mixing ratios and source count. Furthermore, we investigate the suitability of Kernel Additive Modeling (KAM), a technique that can be used for splitting a music recording into harmonic and percussive components.
Concluding the second part of this thesis, we describe a novel method for harmonic-percussive source separation (HPSS) that combines the advantages of KAM and NMF. Our core idea is to use KAM as a means to extract initial estimates of the percussive and harmonic part. Subsequently, both parts are jointly refined through NMF. To this end, we propose two soft constraints that can be applied to NMF activations to shape them into either decaying impulses or plateau-like note activities. Source separation experiments show that this method yields competetive results in comparison to previously proposed HPSS methods.
In the third part of the thesis, we turn to applications of drum-centric music processing for musicological studies. In particular, we focus on the swing ratio, a microrhythmic phenomenon which plays an important role in understanding ensemble performances in jazz music. In Chapter 9, we introduce a formal description of the swing ratio. In baseline experiments, we find that onsets of the ride cymbal in jazz recordings can be robustly detected with simple ADT methods. However, we also show that the mapping from discrete onset times to swing ratios is problematic and can be facilitated by using the log-lag autocorrelation function (LLACF). In this context, the LLACF serves as a mid-level representation of the ride cymbal's most salient periodicities. Based on these findings, we extend the concept of swing ratio estimation by LLACF pattern matching to the swingogram mid-level representation in Chapter 10. We then apply this novel music representation to shed some light on interesting micro-rhythmic phenomena in jazz research.
Finally, we conclude this thesis in Chapter 11 by giving an outlook on future challenges. We use this opportunity to reflect on current shortcomings of the methods studied in this thesis and to formulate our vision of incorporating promising new techniques. We also sketch further application scenarios that can benefit from the research in this thesis.

1.2 Contributions

The main contributions of this thesis can be summarized as follows.

- A detailed overview of methods for ADT that have been published over the last 15 years (Chapter 2, Section 2.5).

- A systematic performance evaluation and comparison of NMF-based versus RNN-based families of ADT approaches (Chapter 3, Section 3.4).
- A systematic analysis of the influence of score-based and audio-based initialization to NMFD with application to separation of drum-only recordings (Chapter 4, Section 4.3).
- Two extensions of informed NMFD using a cascaded decomposition as well as a dictionary of isolated drum sound events for reducing cross-talk artifacts (Chapter 4, Section 4.4).
- An extension to a score-informed NMF approach that uses dictionaries of isolated note events and spectral envelope transfer to improve separation results (Chapter 5, Section 5.4).
- A time-domain modification of an iterative phase reconstruction algorithm with benefits for restoring transient signal components (Chapter 6, Section 6.3).
- An experimental evaluation of generalized Wiener filtering in the context of different music decomposition scenarios (Chapter 7, Section 7.2).
- A combined method for HPSS leveraging the advantages of KAM and NMF for a high-quality extraction of drum sounds (Chapter 8, Section 8.2).
- Appropriate soft constraints for NMF and NMFD promoting the development of activations that resemble decaying impulses (Chapters 4 and 8, Sections 4.2.3 and 8.2.3).
- A novel mid-level representation for visualizing and tracking swing ratios in music recordings with applications to jazz research (Chapter 10, Section 10.3).

1.3 Main Publications

Major parts of this thesis have been previously published in conference proceedings and journal articles in the field of audio signal processing and music information retrieval. In the following, the main publications are listed in chronological order.

[37] Christian Dittmar and Daniel Gärtner. Real-time transcription and separation of drum recordings based on NMF decomposition. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 187–194, Erlangen, Germany, September 2014.

[42] Christian Dittmar, Jonathan Driedger, and Meinard Müller. A separate and restore approach to score-informed music decomposition. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, October 2015.

[44] Christian Dittmar, Martin Pfleiderer, and Meinard Müller. Automated estimation of ride cymbal swing ratios in jazz recordings. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 271–277, Málaga, Spain, October 2015.

[38] Christian Dittmar and Meinard Müller. Towards transient restoration in score-informed audio decomposition. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 145–152, Trondheim, Norway, December 2015.

[39] Christian Dittmar and Meinard Müller. Reverse engineering the Amen break – score-informed separation and restoration applied to drum recordings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(9):1531–1543, 2016.

[46] Christian Dittmar, Jonathan Driedger, Meinard Müller, and Jouni Paulus. An experimental approach to generalized Wiener filtering in music source separation. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Budapest, Hungary, August 2016.

[48] Christian Dittmar, Martin Pfleiderer, Stefan Balke, and Meinard Müller. A swingogram representation for tracking micro-rhythmic variation in jazz performances. *Journal of New Music Research*, 47(2):97–113, 2018.

[47] Christian Dittmar, Patricio López-Serrano, and Meinard Müller. Unifying local and global methods for harmonic-percussive source separation. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Calgary, Canada, April 2018.

[223] Chih-Wei Wu, Christian Dittmar, Carl Southall, Richard Vogl, Gerhard Widmer, Jason Hockman, Meinard Müller, and Alexander Lerch. A review of automatic drum transcription. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(9):1457–1483, 2018.

1.4 Additional Publications

The following publications are also related but not considered in this thesis.

[45] Christian Dittmar, Thomas Prätzlich, and Meinard Müller. Towards cross-version singing voice detection. In *Proceedings of the Deutsche Jahrestagung für Akustik (DAGA)*, pages 1503–1506, Nürnberg, Germany, March 2015.

[43] Christian Dittmar, Bernhard Lehner, Thomas Prätzlich, Meinard Müller, and Gerhard Widmer. Cross-version singing voice detection in classical opera recordings. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 618–624, Málaga, Spain, October 2015.

[147] Meinard Müller, Thomas Prätzlich, and Christian Dittmar. Freischütz Digital – When computer science meets musicology. In Kristina Richts and Peter Stadler, editors, *Festschrift für Joachim Veit zum 60. Geburtstag*, pages 551–573, München, Germany, 2016. Allitera.

[136] Patricio López-Serrano, Christian Dittmar, Jonathan Driedger, and Meinard Müller. Towards modeling and decomposing loop-based electronic music. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 502–508, New York City, USA, August 2016.

[3] Stefan Balke, Jakob Abeßer, Jonathan Driedger, Christian Dittmar, and Meinard Müller. Towards evaluating multiple predominant melody annotations in jazz recordings. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 246–252, New York City, USA, August 2016.

[4] Stefan Balke, Christian Dittmar, Jakob Abeßer, and Meinard Müller. Data-driven solo voice enhancement for jazz music retrieval. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 196–200, New Orleans, Louisiana, USA, March 2017.

[137] Patricio López-Serrano, Christian Dittmar, and Meinard Müller. Mid-level audio features based on cascaded harmonic-residual-percussive separation. In *Proceedings of the Audio Engineering Society Conference on Semantic Audio (AES)*, pages 32–44, Erlangen, Germany, June 2017.

[138] Patricio López-Serrano, Christian Dittmar, and Meinard Müller. Finding drum breaks in digital music recordings. In *Proceedings of the International Symposium on Computer Music Multidisciplinary Research (CMMR)*, pages 68–79, Porto, Portugal, September 2017.

[5] Stefan Balke, Christian Dittmar, and Meinard Müller. Ansätze zur datengetriebenen Transkription einstimmiger Jazzsoli. In *Proceedings of the Deutsche Jahrestagung für Akustik (DAGA)*, pages 1530–1532, München, Germany, March 2018.

[30] Estefanía Cano, Christian Dittmar, Jakob Abeßer, Christian Kehling, and Sascha Grollmisch. Music technology and education. In Rolf Bader, editor, *Springer Handbook on Systematic Musicology*, pages 855–871. Springer, Berlin, Heidelberg, 2018. ISBN 978-3-662-55002-1.

1.5 Acknowledgments

This thesis reflects the results of my work and research conducted in the group of Prof. Dr. Meinard Müller over the last four years. During the latter half of this period I received funding from the DFG project SeReCo[4] (DFG MU 2686/10-1) for which I would like to thank the German Research Foundation.

I conducted my PhD studies at the International Audio Laboratories Erlangen[5] which is a very renowned institution in the field of audio signal processing, and offers an outstanding ecosystem for conducting long-term applied research.

At this point, I would like to take the opportunity to express my thanks to a number of collaborators, colleagues, and friends. First of all, I want to thank Meinard Müller for offering me the unique opportunity to come to Erlangen and pursue my PhD within his group. I strongly benefited from working with him and all the other group members: Stefan Balke, Christof Weiß, Patricio López-Serrano, Frank Zalkow, Vlora Arifi-Müller, Sebastian Rosenzweig, Jonathan Driedger and Thomas Prätzlich. Of course, the great working environment and inspiring research atmosphere in the AudioLabs would not be possible without the administrative staff, the professors and all other colleagues.

I would like to thank Kazuyoshi Yoshii and Frank Kurth for immediately agreeing to review my thesis.

Throughout the PhD, I had some very fruitful scientific collaboration with a number of researchers

[4] Separation and Reconstruction of Drum Sound Components

[5] The International Audio Laboratories Erlangen are a joint institution of the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) and Fraunhofer IIS.

from other institutions: Martin Pfleiderer and Klaus Frieler from the Jazzomat Project, HfM Weimar; Bernhard Lehner, Richard Vogl, and Gerhard Widmer from JKU, Linz; Chih-Wei Wu and Alexander Lerch from Georgia Tech, Atlanta; Carl Southall and Jason Hockman from DMT Lab, Birmingham.
Furthermore, I had the opportunity to focus on investigating phase reconstruction algorithms during a three-month internship at the Fraunhofer IIS. To this end, I would like to thank Jürgen Herre, Sascha Disch, Andreas Niedermayer, and Richard Füg. I also appreciate Jouni Paulus' early works on automatic drum transcription. Since he is now working at IIS, we had the chance to collaborate on generalized Wiener filtering.
Moreover, I want to thank my former colleagues from Fraunhofer IDMT for keeping the friendship going: Hanna Lukashevich, Estefanía Cano, Sascha Grollmisch, Daniel Gärtner, Jakob Abeßer, and many others.
Last but not least, I want to thank my family who unconditionally supported my decision to start a PhD. In particular, I want to thank my beloved wife Cornelia and our lovely daughter Natalie for being with me and bearing with me.

Part I

Automatic Drum Transcription

Chapter 2

An Overview of Automatic Drum Transcription

The work in this chapter is mainly based on the manuscript for [223], resulting from a collaboration between Chih-Wei Wu, Carl Southall, Richard Vogl, Gerhard Widmer, Jason Hockman, Alexander Lerch, Meinard Müller, and myself. CW and myself contributed equally to the literature overview.

In the research area of music information retrieval (MIR), Automatic Music Transcription (AMT) is considered to be one of the most challenging tasks [13]. In simple terms, transcription can be understood as the counterpart of music making. Instead of having musicians perform with their instruments according to a notated sheet music, AMT aims at deriving such symbolic notation from previously recorded music. In the MIR literature, the majority of authors focus on transcribing the fundamental frequency, onset time, and duration of note sequences that are played by pitched instruments such as piano and guitar, or performed by the human singing voice [178]. Usually, these instruments contribute to the harmonic and melodic content of music. In contrast, drums and percussion play an important role for emphasizing and shaping the rhythm. Over the last two decades, several authors have proposed algorithms for ADT, where the equivalent of discerning musical pitches is the detection and classification of drum sound events. More than ten years ago, FitzGerald and Paulus [73] provided a coherent summary of early works on ADT. Compared to the wealth of papers on general AMT topics, one could say that ADT research has gone through hibernation since then. Motivated by recently growing interest in the topic, this chapter presents an updated overview of ADT research. It includes a thorough discussion of the task-specific challenges (Sections 2.1 to 2.4) and a new categorization scheme for existing techniques (Sections 2.5 and 2.6). For a compact overview, Table 2.1 provides a reference for common acronyms and abbreviations while Table 2.2 provides an exhaustive list

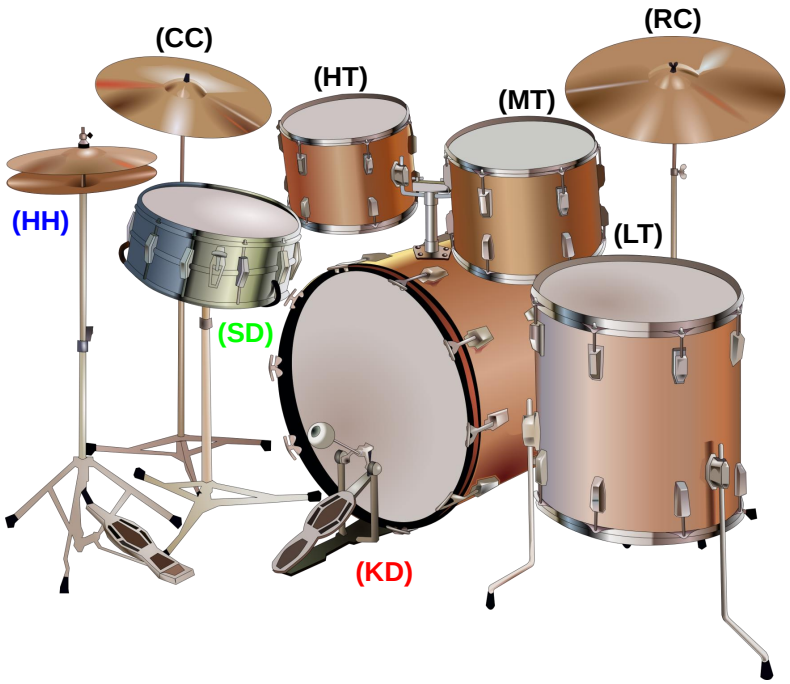

Figure 2.1. The most important parts of a drum kit as used in Western popular music. Kick drum **(KD)**, snare drum **(SD)**, high tom **(HT)**, mid tom **(MT)**, low tom **(LT)** hi-hat **(HH)**, crash cymbal **(CC)**, and ride cymbal **(RC)**. The color-coding and abbreviations are used throughout the thesis.

and categorization of previous publications. Section 2.7 concludes this chapter with some central findings from our literature review.

2.1 Introduction to Drum Kits

The drum kit plays an important role in many Western music genres such as rock, pop, jazz, and dance music. The traditional role of drums in these music genres is to emphasize the rhythmic structure as well as to support the segmentation of the piece into different parts. Generally speaking, the sound characteristics of drum instruments (unpitched, inharmonic, percussive, and transient) differ in many aspects from pitched instruments. It should be noted that there are numerous exceptions to this tendency. For example, there are pitched percussion instruments such as the vibraphone. Moreover, certain instruments such as piano and guitar also comprise transient sound components.

Figure 2.1 introduces the parts of a basic drum kit with their abbreviations and color coding as used throughout this thesis. The different drum instruments can be roughly classified into the two classes membranophones and idiophones. The kick drum (KD), snare drum (SD), high tom (HT), mid tom (MT), and low tom (LT) are typical examples of membranophones, which have vibrating membranes spanned over cylindric bodies. In contrast, the hi-hat (HH), crash cymbal

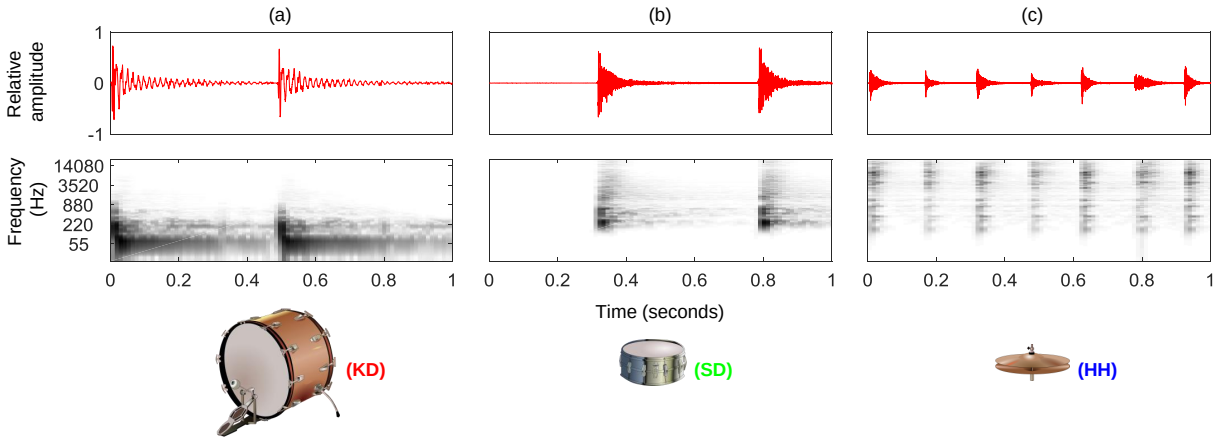

Figure 2.2. Illustration of typical drum sound events of **(a)** kick drum, **(b)** snare drum, and **(c)** hi-hat. The panels show the time-domain signal in red and the corresponding spectrogram representation, with darker shades of gray encoding higher energy. For the sake of visibility, the spectrograms are given with a logarithmically spaced frequency axis and logarithmically compressed magnitude values. Note that we re-use the color-coding and abbreviations introduced in Figure 2.1.

(CC), and ride cymbal (RC) are typical examples of idiophones, whose metallic body vibrates as a whole. It should be noted that the number and variety of toms and cymbals may vary greatly across different music genres (one or two in jazz; up to ten in rock / solo drumming).

Figure 2.2 illustrates the typical sound events produced by the three drum instruments KD, SD, and HH. The KD is played via a foot pedal, generating sounds with low, indefinite pitch. In Figure 2.2a, this can clearly be seen by the concentration of energy in the lower frequency region. In the frequency band around 55 Hz, the initial transient is followed by a slow decay spread over several hundred milliseconds. Depending on the music style and recording conditions, the presence of such tonal components within drum sounds is not an uncommon phenomenon. The SD often acts as the rhythmic counterpart of the KD. It has metallic snare wires stretched across the lower drum head. When striking the upper head with a drum stick, the lower head's vibrations excite the snares, generating a bright sound. In Figure 2.2b, it can be seen that the SD tends to decay faster than the KD and usually covers the middle to high frequency range. The sound of a HH can be influenced and produced by opening or closing it with a foot pedal. When being closed, it produces a quickly decaying clicking sound. When open, it produces a standard cymbal sound exhibiting many inharmonic partials. As shown in Figure 2.2c, the HH's sound components are usually concentrated in the higher frequency regions but often show considerable overlap with the frequency range covered by the SD.

Acoustic drum instruments can produce a wide variety of drum sounds depending on many influences (e.g., the striking position and velocity). Professional drummers may use this freedom for artistic expression. In contrast, drum sampler software usually features a limited number of pre-recorded drum sounds per instrument. To emulate the variability of acoustic drums, it is

common to switch between different samples of the same drum, either based on velocity levels or random selection.

2.2 Challenges and Particularities

As already indicated, drum instruments are quite different from pitched instruments. Hit with sticks or mallets, drums usually start with transient-like sound components exhibiting broadband, noise-like spectra. Tonal components may also occur for certain drum types and playing techniques. Contrasting pitched instruments, the tonal elements are usually not structured like partials in the harmonic series. Instead, their frequency relationship can range from inharmonic to chaotic. This can be seen in Figure 2.2, where the magnitude spectrograms do not exhibit clear harmonic structures. Due to these characteristics, certain algorithms tailored to pitched instruments (e. g., fundamental frequency estimation) are not applicable for ADT. Moreover, variability within the sounds of one drum instrument usually relates to the timbre (i. e., the strength of partials) whereas the partial frequencies change only subtly. For that reason, template-based approaches are often used in ADT.
In music recordings, drum sounds are usually superimposed on top of each other, i.e., different drum instruments are played simultaneously. To illustrate the implications, we show the spectrogram of a funk drum pattern played with KD, SD, and HH in Figure 2.3a. At first sight, one can observe a fairly complex mixture of different drum sound events. As emphasized by the color-coding in Figure 2.3b, there is a strong overlap between KD, SD, and HH in both time and frequency. This can lead to ambiguous situations where it is hard to automatically classify drum sound events or combinations thereof. This challenge is further intensified when drums are mixed with other instruments. There are, however, other properties of the drum signals that can be exploited. For instance, drums are usually played in a locally periodic and repetitive fashion in order to emphasize the meter and pulse. Thus, many onset events of the same drum instrument can be observed throughout a recording, often repeating a rhythmic pattern. This can be utilized by methods that inherently capture these quasi-periodic characteristics.

2.3 Task Definition

In this section, we describe various related tasks that are typically subsumed under the umbrella term ADT. The top-most rows of Table 2.1 gives a compact overview of these tasks. In its most basic form, Drum Sound Classification (DSC) aims at automatic instrument classification of isolated drum sounds. A related task is Drum Sound Similarity Search (DSSS), where the aim is to quantify how similar isolated drum sounds are to each other. Drum Technique Classification

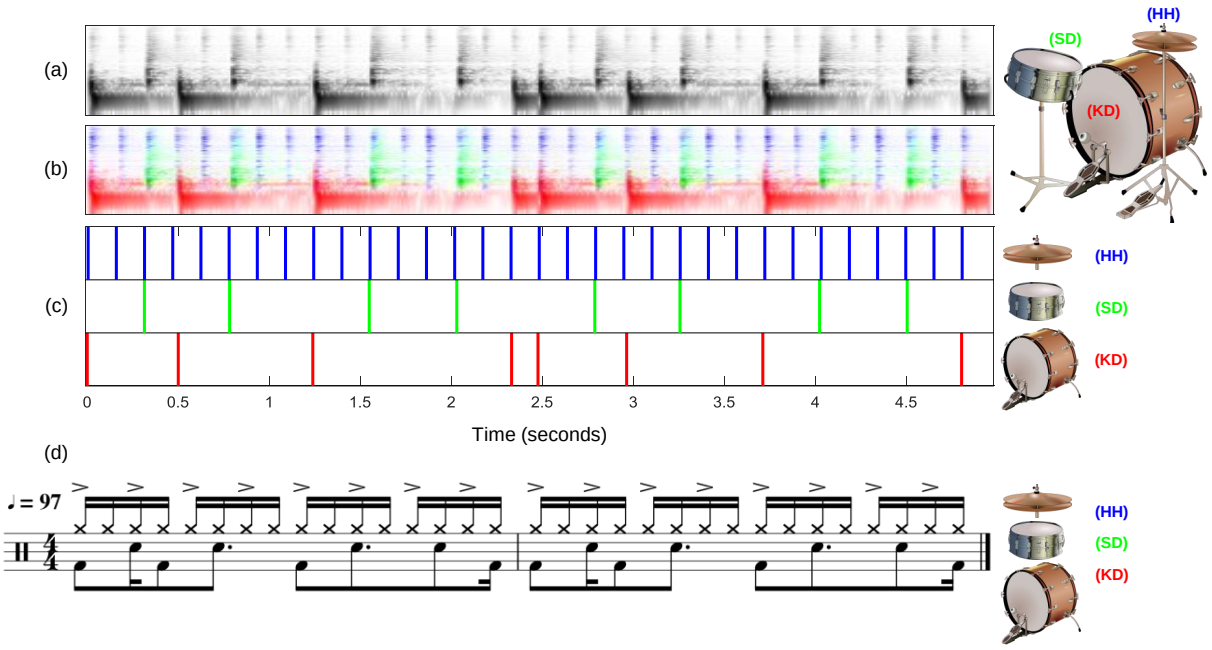

Figure 2.3. Illustration of drum transcription in drum-only recordings. **(a)** Example drum recording in a spectrogram representation with logarithmic frequency spacing and logarithmic magnitude compression. Darker shades of gray encode higher energy. **(b)** The same spectrogram representation with color-coded contributions of individual drum instruments. **(c)** Target onsets displayed as discrete activation impulses. **(d)** Drum notation of the example signal. The note symbols have been roughly aligned to the time axis of the figures above.

(DTC) goes beyond that, paying extra attention to recognizing special playing techniques.

As opposed to isolated drum events, typical drum recordings are sequences of drum sounds. One special case of transcribing a sequence of non-overlapping drum events is voice percussion transcription (VPT), which involves the detection of the percussion sounds produced during beat boxing (a vocal technique to mimic drum rhythms). Due to limitations of the human vocal tract, the drum-like sound events are usually monophonic in the sense that two voice percussion sounds mimicking different drum instruments hardly co-occur.

Drum Transcription of Drum-only recordings (DTD) is a well-studied task that is addressed in many publications. In contrast to VPT, different drum sounds may occur simultaneously, making it more difficult to unambiguously discern multiple drum instruments. A typical output of DTD is shown in Figure 2.3c, where the discrete onsets (i. e., the physical time when a certain drum is hit) are encoded as impulse-like activations. Drum Transcription in the presence of Percussion (DTP) allows that additional percussion instruments besides the targeted ones may be played. Clearly, this is a more complex scenario which typically leads to more erroneously detected onsets. Finally, Drum Transcription in the presence of Melodic instruments (DTM) aims at detecting and classifying the occurrences of different drum sounds in full-mixture music such as pop, rock, or jazz recordings.

We would like to point out that we use the term transcription in a rather loose way, as is common

in the MIR literature. A complete transcription would require to fit the recognized drum onsets into a metrical grid in order to generate a musical score in drum notation. Additionally, other meta data included in sheet music (e. g., instructions for playing techniques, embellishments, indications for tempo and dynamics changes) may be regarded to be part of full transcripts. This is illustrated in Figure 2.3d, where we show the ground-truth drum notation of the example signal. To make the correspondences more obvious, we roughly aligned the musical time axis to the physical time axis of the panels above. For the sake of consistency with prior work, we use the term *drum transcription* in this thesis to cover the detection and classification of drum sound events.

2.4 Application Scenarios

In the following, we want to briefly introduce a few application scenarios that benefit from reliable ADT techniques.

Music Education: Music education software and video games such as RockSmith[6], Yousician[7], or Songs2See[8] could potentially benefit from automatic drum transcription. Very few educational applications offer the possibility to practice drums by using electronic drum pads that output MIDI signals. None of the existing applications allow users to practice on acoustic drum kits. In this context, the goal would be to monitor the players while they are practicing and provide automatic performance assessment, ideally in real-time.

Music Production: In professional music production, drum parts are usually recorded using multiple microphones. Post-processing typically includes equalization, reverberation, dynamics processing, or even drum replacement using specialized plug-ins.[9] It is difficult to properly set up drum microphones and engineer the microphone signals to minimize cross-talk (leakage). In [122], an approach for drum leakage suppression was proposed (which later went into the product Drumatom[10]). With the availability of affordable, easy-to-use, and high-quality drum sample software, it becomes more and more common in music productions to use both sampled drums and recorded acoustic drums with extracted triggers. Having a reliable ADT method at hand would facilitate both drum leakage suppression as well as drum replacement applications.

Music Remixing: Yoshii et al. [225] proposed a drum equalizer for attenuating or replacing drum sounds in complete music recordings. This functionality is enabled by their ADT

[6] `http://rocksmith.ubi.com/`, last accessed June 14, 2018
[7] `https://yousician.com/`, last accessed June 14, 2018
[8] `http://www.songs2see.com/`, last accessed June 14, 2018
[9] `http://www.drumagog.com/`, last accessed June 14, 2018
[10] `http://drumatom.com/`, last accessed June 14, 2018

Category	Acronym	Abbreviation for
Drum Transcription Task	DSC	Drum Sound Classification
	DSSS	Drum Sound Similarity Search
	DTC	Drum Technique Classification
	DTD	Drum Transcription of Drum-only Recordings
	DTP	Drum Transcription in the Presence of Additional Percussion
	DTM	Drum Transcription in the Presence of Melodic Instruments
	OD	Onset Detection
	VPT	Voice Percussion Transcription
Sub-Task	MC	Multi-channel
	RT	Real-time
Feature Representation	AVF	Audio-Visual Features
	BPF	Bandpass Filterbank
	CQT	Constant-Q Transform
	DWT	Discrete Wavelet Transform
	HPSS	Harmonic-Percussive Source Separation
	LLF	Low-Level Audio Features
	LSF	Line Spectral Frequencies
	MLS	Mel-Scale Log Magnitude
	MFCC	Mel-Frequency Cepstral Coefficients
	STFT	Short-Time Fourier Transform
	WAV	Waveform
	ZCR	Zero-Crossing Rate
Method for Activation Function and Feature Transformation	AdaMa	Template Adaptation and Matching
	CNMF	Convolutive NMF
	CFS	Correlation-Based Feature Selection
	FDA	Fisher Discriminant Analysis
	ICA	Independent Component Analysis
	IGR	Information Gain Ratio
	ISA	Independent Subspace Analysis
	LDA	Linear Discriminant Analysis
	MDS	Multi Dimensional Scaling
	MPSC	Matching Pursuit Using Sparse Coding Dictionary
	NSP	Noise Subspace Projection
	NMF	Non-Negative Matrix Factorization
	NMFD	Non-Negative Matrix Factor Deconvolution
	NNICA	Non-Negative ICA
	PCA	Principal Component Analysis
	PFNMF	Partially-Fixed NMF
	PLCA	Probabilistic Latent Component Analysis
	PSA	Prior Subspace Analysis
	RFE	Recursive Feature Elimination
	SANMF	Semi-Adaptive NMF
	SIPLCA	Shift-Invariant PLCA
Classifiers for Frame-Wise Processing	ALC	Alternate Level Clustering
	DNN	Deep Neural Network
	DT	Decision Tree Classifier
	HCA	Hierarchical Cluster Analysis
	KNN	K-Nearest Neighbor Classifier
	SVM	Support Vector Machine
Classifiers Exploiting Temporal Context	BLSTM	Bidirectional LSTM
	BRNN	Bidirectional RNN
	CNN	Convolutional Neural Network
	GRU	Gated Recurrent Unit
	HMM	Hidden Markov Model
	LSTM	Long-Short Term Memory
	RNN	Recurrent Neural Network

Table 2.1. List of acronyms and abbreviations used in this thesis. We have grouped the diverse terms into main categories for better structuring.

algorithm that detects prominent drum sound events and estimates their timbre. Dittmar and Müller [39] showed that reliable drum transcription is beneficial for decomposing monaural drum recordings into single drum hits for the purpose of remixing. In this context, a score-informed audio-aligned transcription is used for initialization of an audio decomposition method. Recently, the music software Regroover, whose main feature is a similar source separation technology, was released[11]. For certain tasks, this software still requires a lot of intervention by the user, which could be alleviated when having a reliable ADT algorithm at hand.

Music Information Retrieval: More generally speaking, ADT is a useful preprocessing step for obtaining high-level music content descriptions in MIR. First, transcription of the drums is an important prerequisite for determining the rhythm patterns. It was shown that this is an important aspect for structuring large corpora of popular music [167] as well as electronic music [131]. Going towards musicological research, there is a high interest in determining microrhythmic properties such as swing, shuffle and groove [35, 44, 48] inherent in music recordings. ADT can be beneficial for such analyses as well.

2.5 General Approaches to Drum Transcription

In this section, we summarize important concepts in ADT research. FitzGerald and Paulus [73] proposed to categorize the systems into two types, namely the *pattern recognition* and the *separation-based* approaches. Later on, a more refined grouping into four categories was proposed by Gillet and Richard [92] and taken up by Paulus [158]. These approaches are referred to as *Segment and Classify*, *Separate and Detect*, *Match and Adapt*, and *HMM-based Recognition*. Although we highly appreciate the work of these authors, we found that this classic categorization does not accurately reflect the advances in ADT published in recent years.
In this chapter, we propose a new grouping according to six generic algorithmic concepts, see Figure 2.4 for an overview. Before we briefly introduce each of these concepts in the following paragraphs, we first want to emphasize that they can be used like items from a toolbox, interchangeably and in no particular order. Second, the distinction between the concepts is sometimes vague. And third, the concepts are often not specific to drum or music recordings, but very generic, e. g., inspired from research in speech, language, and general multimedia processing.

Feature Representation (FR): Apart from the time-domain waveform, discretized audio signals can also be converted into feature representations that are better suited for certain processing tasks. A natural choice are Time-Frequency (TF) transforms (e. g., Short-Time Fourier Transform, STFT), or Low-Level Features (LLF), e. g., Mel-Frequency Cepstral Coefficients (MFCC) derived

[11] `http://regroover.com/`, last accessed June 14, 2018

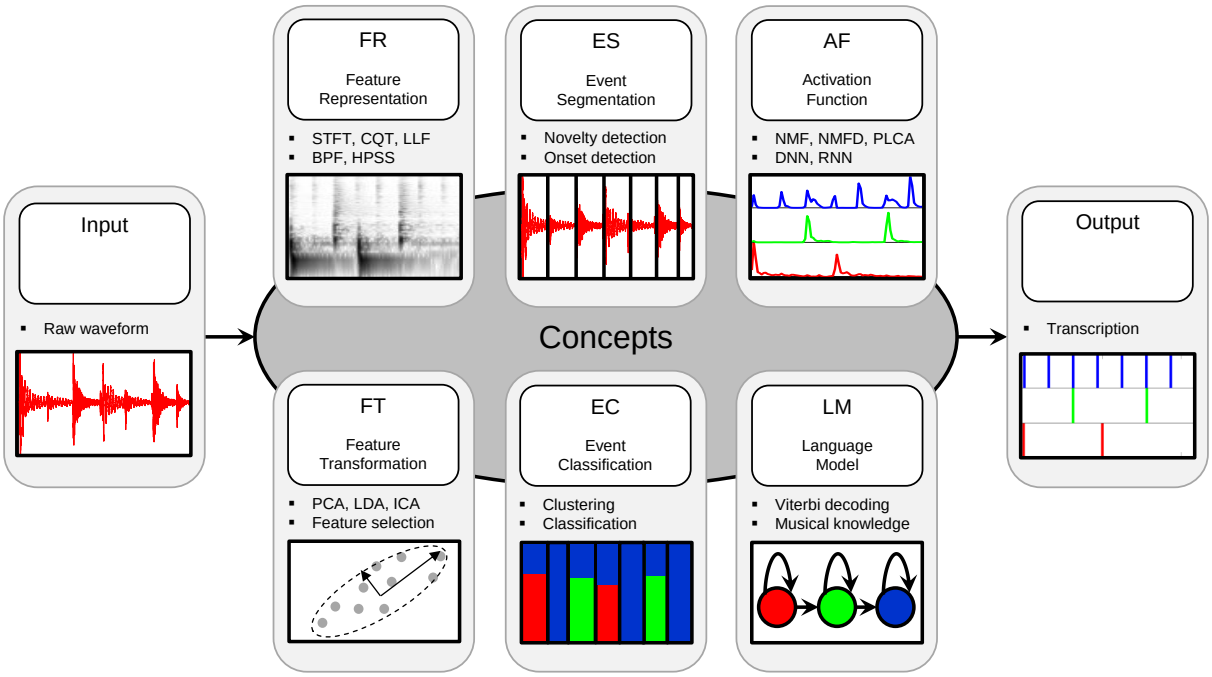

Figure 2.4. Our proposed grouping of algorithmic concepts that are relevant for ADT.

from them. These representations are beneficial for untangling and emphasizing the important information hidden in the audio signal. We also subsume processing steps intended to emphasize the target drum signal in an audio mixture to belong to this concept. These can either be based on spectral characteristics (e.g., band-pass filters, BPF, with pre-defined center frequencies and bandwidths) or based on TF characteristics (e. g., harmonic-percussive source separation, HPSS).

Feature Transformation (FT): This concept provides a transformation of the feature representation to a more compact form. This goal can be achieved by different techniques such as feature selection, Principal Component Analysis (PCA), or Linear Discriminant Analysis (LDA). It should be mentioned that there is a strong overlap between the concepts **FT** and **AF** (as discussed below). **FT** usually serves as a post-processing step for **FR** and arrives at a more compact feature representation, whereas **AF** is specifically used for converting the signal into drum-related activation functions. Still, **AF** techniques can also be used for **FT** purposes.

Event Segmentation (ES): The main goal of this concept is to detect the temporal location of musical events in a continuous audio stream before applying further processing. This usually consists of computing suitable novelty functions (e.g., Spectral Flux) and identifying locations of abrupt change. A typical procedure would be to extract local extrema by applying a suitable peak-picking strategy, often referred to as OD in MIR research. Recently, learned feature representations have been shown to yield superior performance compared to hand-crafted ones for

event segmentation.

Event Classification (EC): This processing step aims at associating the instrument type (e. g., KD, SD, or HH) with the corresponding musical event. In the majority of papers, this is achieved through machine learning methods (e.g., Support Vector Machines, SVM) that can learn to discriminate the target drum instruments (or combinations thereof) based on training examples. Inexpensive alternatives include clustering (e.g., Alternate Level Clustering, ALC) and cross-correlation.

Activation Function (AF): This concept seeks to map feature representations into activation functions, which indicate the activity of different drum instruments over time. Different techniques such as NMF, Probabilistic Latent Component Analysis (PLCA) or Deep Neural Networks (DNN) are commonly used for deriving the activation functions. As already indicated, **AF** overlaps considerably with the concept **FT** introduced above, especially in the sense that technically some sort of transform (e. g., affine transforms) is used to convert an **FR** to a desired activation function. The main difference is that the target is not to compress **FR** w.r.t. some optimality criterion but rather to reveal latent variables that help to discriminate the target drum instruments.

Language Model (LM): This concept takes the sequential relationship between musical events into account. Usually, this is achieved using a probabilistic model capable of learning the musical grammar and inferring the structure of musical events. **LM**s are based on classical methods such as Hidden Markov Models (HMM) or more recently popularized methods such as Recurrent Neural Networks (RNN).

2.6 Literature Overview

While the algorithmic concepts introduced above represent essential building blocks, usually only a subset of them are used in specific ADT approaches. One common pattern is to use the algorithmic concepts in a cascaded fashion, often in processing chains comprising three to four blocks. Since this gives rise to many possible configurations, it is more appropriate to present the literature overview in tabular form instead of textual form. Thus, Table 2.2 summarizes the most important details. The first two columns provide the year, author and literature reference. The third column lists the specific ADT task, the fourth column gives the **FR** type used in the respective papers. The penultimate column of Table 2.2 presents the specific cascade of concepts,

Year	References	ADT Task (Sub-Task)	Feature Representation	Cascade of Algorithmic Concepts	Core Method
1985	Schloss [180]	DTD	WAV	**FR**, **ES**	OD
2000	Gouyon et al. [95]	DSC	LLF	**FR**, **EC**	FDA
2002	FitzGerald et al. [74]	DTD	STFT	**FR**, **AF**, **ES**	ISA
2002	Herrera et al. [105]	DSC	LLF	**FR**, **ES**, **FT**, **EC**	KNN
2002	Zils et al. [230]	DTM	WAV, ZCR	**ES**, **EC**	ABS
2003	Eronen [65]	DSC	STFT	**FR**, **ES**, **FT**, **EC**	ICA & HMM
2003	FitzGerald et al. [70, 75, 76]	DTD	STFT	**FR**, **AF**, **ES**	PSA
2003	Herrera et al. [106]	DSC	LLF	**FR**, **ES**, **FT**, **EC**	DC & MDS
2004	Dittmar and Uhle [40]	DTM	STFT	**FR**, **AF**, **ES**	NNICA
2004	Gillet and Richard [87]	DTD	LLF, MFCC	**FR**, **FT**, **LM**	HMM & SVM
2004	Herrera et al. [107]	DSC	LLF	**FR**, **ES**, **FT**, **EC**	KNN
2004	Nakano et al. [149]	VPT	MFCC	**FR**, **LM**	HMM
2004	Sandvold et al. [179]	DTM	LLF	**FR**, **ES**, **FT**, **EC**	DT
2004	Steelant et al. [193, 194]	DSC	LLF	**FR**, **EC**	SVM
2004	Tindale et al. [203]	DTC	LLF	**FR**, **FT**, **EC**	SVM & KNN
2004	Yoshii et al. [224, 226, 227]	DTM	STFT	**FR**, **ES**, **EC**	AdaMa
2005	Degroeve et al. [36]	DTM	LLF	**FR**, **ES**, **FT**, **EC**	SVM
2005	Gillet and Richard [88]	DTM	BPF	**FR**, **ES**, **FT**, **EC**	NSP
2005	Gillet and Richard [89]	DTM	AVF	**FR**, **ES**, **FT**, **EC**	SVM
2005	Hazan [101]	VPT	LLF	**FR**, **ES**, **EC**	DT, KNN
2005	Paulus and Virtanen [160]	DTD	STFT	**FR**, **AF**, **ES**	NMF
2005	Tanghe et al. [200]	DTM	LLF	**FR**, **ES**, **FT**, **EC**	SVM
2005	Tzanetakis et al. [204]	DTM	DWT	**FR**, **ES**	BPF & OD
2006	Bello et al. [9]	DTD	LLF	**FR**, **ES**, **EC**	HCA
2007	Gillet and Richard [91]	DTM	Symbolic	**FR**, **ES**, **LM**	N-gram
2007	Moreau and Flexer [145]	DTM	LLF	**FR**, **ES**, **FT**, **EC**	KNN
2007	Roy et al. [177]	DSC	LLF	**FR**, **ES**, **FT**, **EC**	SVM
2008	Gillet and Richard [92]	DTM	MFCC	**FR**, **ES**, **FT**, **EC**	SVM
2008	Pampalk et al. [154]	DSS	MLS	**FR**, **ES**, **FT**, **EC**	MNSR
2009	Alves et al. [2]	DTM	STFT	**FR**, **AF**, **ES**	NMF
2009	Paulus and Klapuri [158, 159]	DTM	STFT	**FR**, **FT**, **LM**	NMF & HMM
2010	Scholler and Purwins [181]	DSC	MPSC	**FR**, **FT**, **ES**, **EC**	DT
2010	Spich et al. [192]	DTM	STFT	**FR**, **FT**, **ES**	PSA
2011	Şimşekli et al. [34]	DTD	STFT	**FR**, **LM**	HMM
2011	Scholler and Purwins [182]	DSC	FL	**FR**, **FT**, **ES**, **EC**	SVM
2012	Battenberg et al. [7, 8]	DTD (RT)	STFT	**ES**, **FR**, **AF**	NMF
2012	Kaliakatsos et al. [116]	DTD	WAV	**FR**, **ES**	BPF & OD
2012	Lindsay-Smith et al. [133]	DTD	STFT	**FR**, **AF**, **ES**	NMFD
2013	Miron et al. [143, 144]	DTD (RT)	LLF	**ES**, **FR**, **EC**	KNN
2014	Dzhambazov [58]	DTM	LLF	**FR**, **FT**, **LM**	HMM
2014	Benetos et al. [15]	DTM	CQT	**FR**, **AF**, **ES**	SIPLCA
2014	Dittmar and Gärtner [37]	DTD (RT)	STFT	**FR**, **AF**, **ES**	SANMF
2014	Thompson et al. [201]	DTM	MFCC	**FR**, **ES**, **EC**	SVM
2015	Röbel et al. [174]	DTM	STFT	**FR**, **AF**, **ES**	NMFD
2015	Souza et al. [191]	DSC, DTC	MFCC, LSF	**ES**, **FR**, **EC**	SVM
2015	Rossignol et al. [176]	DTM	LLF	**FR**, **EC**, **ES**	ALC
2015	Wu and Lerch [220, 221]	DTM	STFT	**FR**, **AF**, **ES**	PFNMF
2016	Gajhede et al. [85]	DSC	MLS	**ES**, **FR**, **EC**	CNN
2016	Vogl et al. [212, 213]	DTD, DTM	STFT	**FR**, **AF**, **ES**	RNN
2016	Southall et al. [189]	DTM	STFT	**FR**, **AF**, **ES**	BRNN
2016	Wu and Lerch [222]	DTC	STFT	**FR**, **AF**, **ES**, **EC**	PFNMF & SVM

Table 2.2. Overview of previously proposed methods for ADT. The properties in the columns highlight the most important algorithmic details. For a reference to the acronyms, we refer to Table 2.1.

whereas the last column indicates the core methods used (often from the concepts **FT**, **AF**, and **EC**).

From left to right, the cascade of algorithmic concepts describes in which order the tools introduced in Section 2.5 are applied. Although the concept usage and order seem to be rather

arbitrary on first sight, we can observe certain tendencies in the development of ADT over the last 15 years. Between 2005 and 2008, there seems to be a cluster of works using the concepts **FR**, **ES** as the initial stages of their ADT processing chains. These correspond roughly to the previously established category of *Segment and Classify* methods.
Starting with FitzGerald et al. [74], there have been a number of works using the combination of **FR**, **AF** as the initial building blocks. Previously, these methods were subsumed under the category *Separate and Detect*. Starting with Lindsay-Smith et al. [133], we observe a new cluster of these approaches emerging since 2012. To reflect this resurgence, we introduce the notion of *activation-based* ADT methods here. In this context, activation-based means that methods of this kind aim at extracting **AF** that represent the change of individual drum instruments' intensity over time, ideally shaped like the spiky peaks shown in Figure 2.3c.
In this literature overview, we refrain from reporting performance figures of the different approaches. The wide range of different ADT tasks and the heterogeneous evaluation scenarios complicate any systematic and fair comparison. Just recently, attempts have been made to use common datasets, evaluation strategies, and performance metrics. We will specifically address this issue in the following chapter.

2.7 Conclusions and Further Notes

In this background chapter, we provided an up-to-date overview of scientific works in the field of automatic drum transcription (ADT) that have been published during the last 15 years. This work closes the gap that existed since the previous survey [73], published more than a decade ago. We discussed the particularities of drum sound events and the challenges of the ADT scenario in comparison to transcribing other musical instruments. Furthermore, we sketched typical application scenarios that benefit from ADT research. Going beyond previous works, we proposed a novel categorization scheme consisting of six algorithmic concepts that are frequently employed in ADT system. Based on these concepts, we structured the results of our literature survey in a concise form in Table 2.2.
As already mentioned, one can observe over the last few years that there is a slight tendency towards using the concepts **FR** and **AF** as the two initial processing steps. Methods that follow this paradigm will be referred to as activation-based ADT systems for the remainder of this thesis. Since state-of-the-art ADT results were reported in the literature for activation-based approaches, we are going to investigate these in more detail in the following chapter.

Chapter 3

Activation-Based Drum Transcription Methods

The work in this chapter follows in parts the article [37]. The experimental evaluation is based on the manuscript [223], resulting from a collaboration between Chih-Wei Wu, Carl Southall, Richard Vogl, Gerhard Widmer, Jason Hockman, Alexander Lerch, Meinard Müller, and myself. CS proposed the experimental setup and carried out major parts of the evaluation.

In the previous chapter, we have shown that the literature on ADT is very diverse in terms of the methods and approaches that have been used. To provide more insights on successful ADT systems, we focus in this chapter on two families of activation-based ADT techniques. We start by briefly explaining the methods' technical details and drum-specific modifications. To this end, Section 3.1 introduces the basic mathematical notation that will be used throughout this thesis. This is followed by the description of two specific algorithmic paradigms: Non-Negative Matrix Factorization (NMF) in Section 3.2 and Recurrent Neural Networks (RNNs) in Section 3.3. In parts, these methods will re-appear in subsequent chapters as fundamental tools for drum sound separation tasks.
We then continue by systematically evaluating variants of these basic approaches within a controlled experimental setup using two publicly available datasets. In Section 3.4, we explain the datasets and evaluation strategies that we used to compare NMF-based and RNN-based ADT methods. Furthermore, we present the most important findings from our experiments in condensed form. Section 3.5 concludes this chapter as well as the first part of this thesis.

3.1 Introduction

Following the brief discussion of general ADT approaches in the previous chapter, we now want to focus on two families of ADT algorithms that are currently defining the state-of-the-art, namely NMF-based and RNN-based approaches. The next four sections will provide a more detailed discussion on these techniques. In particular, a comprehensive evaluation will highlight the strengths and weaknesses of the different approaches. In order to put both in a unified perspective, we will start with the introduction of a common signal model.

3.1.1 Common Notation

The following mathematical notation will be used throughout the remainder of this thesis. Uppercase italic letters such as K will be used to denote fixed scalar parameters, while lowercase italic letters such as k are used to denote running variables or indices. We denote integer intervals as $[1:K] := \{1, 2, \ldots, K\}$. Uppercase non-italic letters such as V usually denote matrices, while lower-case non-italic letters such as v denote column vectors. $\mathrm{V}^\top$ denotes the matrix transpose of V. Rounded brackets are used to refer to elements of vectors and matrices, e.g., $\mathrm{V}(k, m)$ refers to the element located at the k^{th} row and the m^{th} column of matrix V. The colon is a short notation for taking slices along a certain dimension of a matrix, e.g., $\mathrm{V}(:, m)$ denotes the m^{th} column. For notational convenience, we also introduce the subscript notation $\mathrm{v}_k := \mathrm{V}(k, :)$ and the superscript notation $\mathrm{v}^m := \mathrm{V}(:, m)$. In Section 3.3, we will make extensive use of that notation, also for the sake of compatibility with previous work. Other notational conventions will be explained in the respective chapters.

3.1.2 Feature Representation

Both families of ADT systems considered here belong to activation-based methods. As such, they are all based on the signal model assumption that the given drum recording is approximately a linear mixture of constituent drum sound events. Let $\mathrm{V} \in \mathbb{R}_{\geq 0}^{K \times M}$ be a matrix-representation of the signal's magnitude spectrogram, with $\mathrm{V}(k, m)$, representing the non-negative, real-valued TF magnitude coefficient at the k^{th} spectral bin for $k \in [0:K]$ and the m^{th} time frame for $m \in [1:M]$. The number of frequency bins is determined by the window size N as $K = N/2$. The number of spectral frames M is determined by the available signal samples. Our objective is to map V to an activation representation $\mathrm{H} \in \mathbb{R}_{\geq 0}^{R \times M}$. Here, the number of rows $R \in \mathbb{N}$ is usually equal to the number of distinct drum instruments (e.g., $R = 3$ for KD, SD, HH). As H encodes the activations of a certain drum instrument over time, $\mathrm{H}(r, :)$ should be large for time

instances where the r^{th} instrument is active (i. e., the instrument is audible). Moreover, it is often desirable that the activations are impulse-like, with their highest peak at the frame that corresponds the onset time (i. e., the time instant when the drum is struck) as shown in Figure 2.3c.

3.1.3 Event Segmentation

As indicated before, the detection of candidate onset events is typically approached by peak-picking in the activation function $\mathrm{H}(r,:)$ for each $r \in [1:R]$. In this chapter, we employ a very simple procedure to locate such peaks. First, a dynamic threshold $\Lambda \in \mathbb{R}_{\geq 0}^{R \times M}$ is calculated for each considered drum instrument and each frame using

$$\Lambda(r,m) = \frac{1}{2\Gamma+1} \sum_{n=m-\Gamma}^{m+\Gamma} \mathrm{H}(r,n). \tag{3.1}$$

Assuming suitable zero padding at the boundaries, $\Gamma \in \mathbb{N}$ determines the window used to calculate the average. Second, we introduce a binary-valued output matrix $\mathrm{O} \in \mathbb{B}^{R \times M}$ with $\mathbb{B} := \{0,1\}$. The elements of O encode onset candidates and are defined as follows:

$$\mathrm{O}(r,m) = \begin{cases} 1, & \mathrm{H}(r,m) = \max\left(\mathrm{H}(r,m-\Omega:m+\Omega)\right) \\ & \text{and} \quad \mathrm{H}(r,m) > \Lambda(r,m) \\ 0, & \text{otherwise} \end{cases} \tag{3.2}$$

Here, $\Omega \in \mathbb{N}$ determines the window used for local maximum search. In other words, a candidate peak is only accepted if it exceeds the dynamic threshold Λ as well as its local neighborhood determined by Ω. If the criterion in (3.2) also is true for two peaks within a certain distance, the weaker of both is discarded. The output matrix O will become important again in the context of evaluation metrics in Section 3.4.3.

3.2 NMF-Based ADT Systems

In this section, we provide more details of ADT systems employing variants of NMF. Figure 3.1 depicts the basic principle of decomposing the mixture's magnitude spectrogram V into spectral basis functions $\mathrm{W}(:,r)$ (called templates), and corresponding time-varying gains $\mathrm{H}(r,:)$ (called activations). Intuitively speaking, the templates comprise the spectral content of the mixture's constituent components, while the activations describe when and with which intensity they occur.

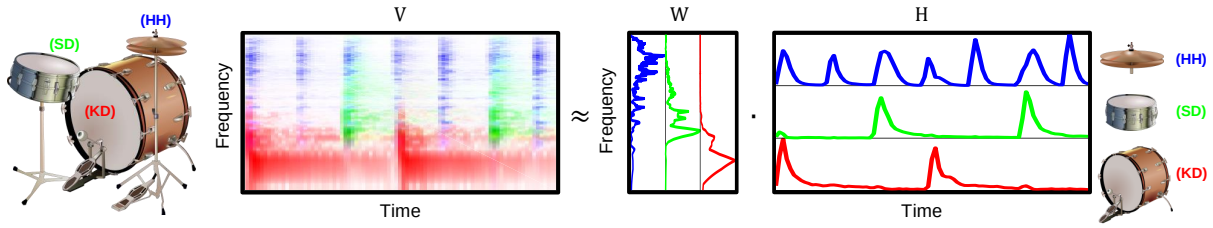

Figure 3.1. Illustration of an NMF-based ADT system. The individual drum instruments appear in the same order as in Figure 2.3.

The entries of template matrix W can be interpreted as averaged spectra of the corresponding drum instruments KD, SD, and HH. The KD in red occupies the lowest frequency region, the SD in green occupies the mid-region, and finally the HH in blue has most of its energy in the upper frequency region. In H, the corresponding drum onset events occur as peaks with quickly rising attacks. They are followed by exponentially decaying slopes that correspond to the natural decay of the drum sound events.

3.2.1 Basic NMF Model

Mathematically, NMF is based on iteratively computing a low-rank approximation $\tilde{\mathrm{V}} \in \mathbb{R}_{\geq 0}^{K \times M}$ of the mixture spectrogram V. Specifically, $\tilde{\mathrm{V}}$ is defined as the linear combination of the templates $\mathrm{W} \in \mathbb{R}_{\geq 0}^{K \times R}$ and activations $\mathrm{H} \in \mathbb{R}_{\geq 0}^{R \times M}$ such that $\mathrm{V} \approx \tilde{\mathrm{V}} := \mathrm{WH}$.
NMF typically starts with a suitable initialization of the matrices W and H. For example, both matrices could be populated with non-negative random numbers. As we will discuss later, more sensible initialization strategies are advised to yield meaningful decomposition results. After initialization, both W and H are iteratively updated to approximate V with respect to a cost function $\mathcal{L}$. A standard choice is the generalized Kullback-Leibler Divergence [130], given as

$$\mathcal{L} = \mathcal{D}_{\mathrm{KL}}(\mathrm{V} \mid \tilde{\mathrm{V}}) = \sum \left(\mathrm{V} \odot \log \left(\frac{\mathrm{V}}{\tilde{\mathrm{V}}} \right) - \mathrm{V} + \tilde{\mathrm{V}} \right). \tag{3.3}$$

The symbol $\odot$ denotes element-wise multiplication; the logarithm and division are to be performed element-wise as well. The sum is to be computed over all KM elements of V. To minimize this cost, an alternating scheme with multiplicative updates is used [130]. The respective update rules are given as

$$\mathrm{W} \leftarrow \mathrm{W} \odot \frac{\frac{\mathrm{V}}{\bar{\mathrm{V}}} \cdot \mathrm{H}^{\top}}{\mathrm{J} \cdot \mathrm{H}^{\top}}, \tag{3.4}$$

$$\mathrm{H} \leftarrow \mathrm{H} \odot \frac{\mathrm{W}^{\top} \cdot \frac{\mathrm{V}}{\bar{\mathrm{V}}}}{\mathrm{W}^{\top} \cdot \mathrm{J}}, \tag{3.5}$$

where the symbol $\cdot$ denotes the matrix product. Furthermore, $\mathrm{J} \in \mathbb{R}^{K \times M}$ denotes a matrix of ones. Since this is an alternating update scheme, it should be noted that Eq. (3.4) uses the latest update of H from the previous iteration. In the same vein, (3.5) uses the latest update of W. These update rules are typically applied for a limited number of iterations $L = L^{\mathrm{NMF}}$, with the iteration index $\ell \in [0 : L]$.

3.2.2 Fixed-Bases NMF (FNMF)

When using NMF for ADT, it is essential to choose a suitable rank $R \in \mathbb{N}$ of the approximation (i.e., number of components) and to provide a good initialization for W. One popular choice (see for example [7, 37, 158, 160]) is to set R to the number of distinct drum instruments and to initialize individual columns $\mathrm{W}(:, r)$ with averaged spectra of isolated drum sound events. The rationale is to let the NMF component updates start from a point in the parameter space that is already close to a meaningful local optimum.
In this context, some authors [7, 37] also propose to keep the initialized $\mathrm{W}(:, r)$ fixed throughout the NMF iterations, i. e., not to apply the update equation (3.4). Although this is a very appealing and simple approach, fixed NMF bases may be problematic in cases where the mixture consists of other components than the previously trained drum sounds. Intuitively speaking, the NMF updates rules will try to model the observed V as accurately as possible given the fixed prior basis vectors, possibly leading to spurious activations that resemble cross-talk between the different drum sounds.

3.2.3 Semi-Adaptive NMF (SANMF)

An alternative approach for combining meaningful initialization with adaptability is to allow the spectral bases in W to deviate from their initial shape with increasing iteration count. Dittmar and Gärtner [37] proposed to enforce this behavior by blending between the latest update of W obtained from (3.4) and the fixed initial dictionary denoted as $\overline{\mathrm{W}}$:

$$\mathrm{W} \leftarrow (1 - \alpha) \cdot \overline{\mathrm{W}} + \alpha \cdot \mathrm{W}. \tag{3.6}$$

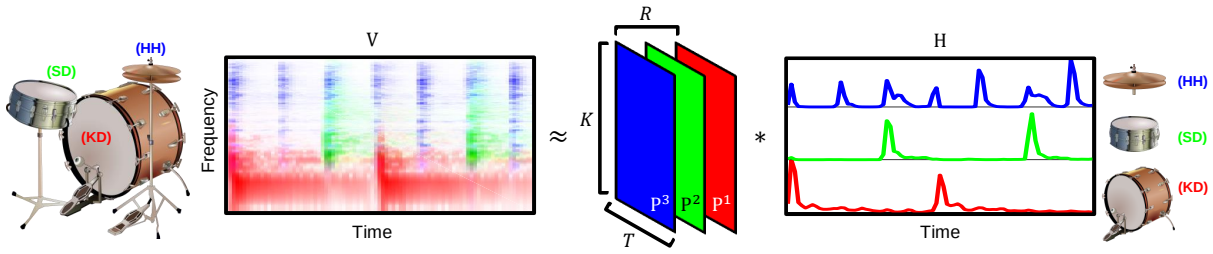

Figure 3.2. Illustration of an NMFD-based ADT system.

The blending parameter α depends on the ratio of the current iteration count ℓ to iteration limit L taken to the power of β:

$$\alpha = \left(\frac{\ell}{L}\right)^{\beta}. \tag{3.7}$$

Thus, only small adaptations of the NMF components are allowed early on, whereas stronger adaptation are allowed in later iterations. The larger the parameter β, the longer one attenuates the influence of the update equation (3.4) on W.

3.2.4 Non-Negative Matrix Factor Deconvolution (NMFD)

The different NMF methods presented so far assumed that one template per drum instrument is sufficient to model temporal dynamics of drum sounds. However, we indicated already in Section 2.1 that certain drum instruments may generate complex, time-varying patterns when being struck. This is in line with the findings of [7, 8], where separate NMF templates for attack and decay of a drum sound event are used.

As an alternative to that, previous works (such as [39, 125, 133, 174, 185]) successfully applied NMFD, a convolutive version of NMF, for drum transcription and source separation. Alternatively, Probabilistic Latent Component Analysis (PLCA) [184] and its convolutive (or shift-invariant) extensions [188] have been used for AMT [13].

As has been discussed in the above-mentioned publications, the NMFD model assumes that all drum sound events occurring in the mixture can be explained by a prototype event that acts as an impulse response to some impulse-like activation (e.g., striking a particular drum). In Figure 3.2, we illustrate this by introducing $R = 3$ prototype magnitude spectrograms $\mathrm{P}^r \in \mathbb{R}_{\geq 0}^{K \times T}$. Each P^r can be directly interpreted as a spectrogram pattern consisting of $T \ll M$ spectral frames. Each pattern is convolved with the corresponding row of H. Superposition of the individual convolution results yields a convolutive approximation of V.

Mathematically, this can be formalized by grouping the above-mentioned patterns into a pattern tensor $\mathrm{P} \in \mathbb{R}_{\geq 0}^{K \times R \times T}$. In short notation, the slice of the tensor which refers to the r^{th} pattern is $\mathrm{P}^r := \mathrm{P}(:, r, :)$; whereas $\mathrm{P}_t := \mathrm{P}(:, :, t)$ refers to the t^{th} frame index in all patterns simultaneously.

The convolutive spectrogram approximation $\mathrm{V} \approx \tilde{\mathrm{V}}$ is modeled as:

$$\tilde{\mathrm{V}} := \sum_{t=0}^{T-1} \mathrm{P}_t \cdot \overset{t\rightarrow}{\mathrm{H}}, \tag{3.8}$$

where $\overset{t\rightarrow}{(\cdot)}$ denotes a frame shift operator. Similar to NMF, both P and H are suitably initialized. Subsequently, they are iteratively updated to minimize a cost function between the convolutive approximation $\tilde{\mathrm{V}}$ and V. According to [185], the update rules extending (3.4) and (3.5) are given by:

$$\mathrm{P}_t \leftarrow \mathrm{P}_t \odot \frac{\frac{\mathrm{V}}{\tilde{\mathrm{V}}} \cdot \left(\overset{t\rightarrow}{\mathrm{H}}\right)^\top}{\mathrm{J} \cdot \left(\overset{t\rightarrow}{\mathrm{H}}\right)^\top} \tag{3.9}$$

$$\mathrm{H} \leftarrow \mathrm{H} \odot \frac{\mathrm{P}_t^\top \cdot \overset{\leftarrow t}{\left[\frac{\mathrm{V}}{\tilde{\mathrm{V}}}\right]}}{\mathrm{P}_t^\top \cdot \mathrm{J}} \tag{3.10}$$

for $t \in [0 : T-1]$.

As can be seen in Figure 3.2, the NMFD-based activations in H exhibit a more spiky, impulse-like shape compared to the ones resulting from NMF in Figure 3.1. As said before, this is a desirable property since it alleviates the **ES** step. The peaks are more concentrated since the P^r have the capability to better model the decay part of the drum sound events, thus attenuating the level of the activations during the decay phase. Learned P^r can be interpreted as a kind of spectrogram representation averaged from all instances of the target drum sound occuring in the signal.

As such, NMFD is conceptually similar to the classic *AdaMa* method [224, 226, 227]. In principle, the typical alternation between drum detection and drum spectrogram template refinement in AdaMa is also reflected in the update rules for NMFD activations and templates. In contrast to AdaMa, no explicit decision-making about the acceptance of drum sound candidates is required during NMFD updates, so that the optimization of some hard thresholds can be omitted.

3.3 RNN-Based ADT Systems

In this section, we provide more details of ADT systems based on recurrent variants of DNNs, called RNNs. Figure 3.3 illustrates the basic concept behind ADT with RNNs. In contrast to the NMF-based systems, the mixture spectrogram V is processed as a time series in a frame-wise fashion, i. e., we insert each individual spectral frame v^m sequentially into a trained RNN. If an input frame corresponds to the start of a drum sound event, it should ideally lead to a spiky,

impulse-like activation at the RNNs' output as shown in Figure 3.3e. In order to explain the necessary training steps enabling this desired input-output behavior, we first review some basics on DNN training.

3.3.1 DNN Training

DNNs are networks consisting of linear operations with learnable parameters (weights and biases) and fixed non-linearities. These essential building blocks are usually organized in layers. For our concrete ADT tasks, we use spectral slices $\mathrm{v}^m := \mathrm{V}(:, m)$, i. e., individual columns of V as input to the first layer. Processing the input data in the first layer is interpreted as transformation to a more abstract representation, which in turn is used as input to the subsequent layer. Ideally, when the data has been processed by all layers, the neurons in the network's output layer should generate activation functions of the assigned drum instruments, as shown in Figure 3.3. This is achieved by training the network with pairs of typical input data and target output data, where the learnable parameters are automatically adjusted towards the desired behavior. In our ADT scenario, the target output is typically generated from ground-truth transcriptions of the training data. For each of the considered drum instruments, frames corresponding to the start of a drum sound event are labeled as 1 and the remaining frames as 0 (as shown in Figure 2.3c). The trained DNN should then produce similar activation functions when provided with the spectrogram input data of previously unseen drum mixtures.
Mathematically, the input-output behavior of a single network-layer can be formalized as

$$\mathrm{h} = \phi\left(\mathrm{W} \cdot \mathrm{v} + \mathrm{b}\right), \tag{3.11}$$

where $\mathrm{W} \in \mathbb{R}^{D \times K}$ is the weight matrix and $\mathrm{b} \in \mathbb{R}^D$ is the bias vector. The non-linearity $\phi(\cdot)$ is applied in an element-wise fashion to yield the layers' output $\mathrm{h} \in \mathbb{R}^D$. A variety of non-linearities are used in the literature, the most common ones being hyperbolic tangent (tanh), sigmoid (σ), and rectified-linear units (ReLU). The meta-parameter $D \in \mathbb{N}$ determines the number of neurons in a given layer and is also referred to as layer width. Sticking to our ADT example of detecting KD, SD, HH sound events using just a single network layer, $D = 3$ would be a natural choice. In multi-layer networks, the hidden layers preceding the output are usually much wider (see Table 3.1).
In accordance to the literature, we denote the entirety of network parameters as the set Θ, such that $\mathrm{W} \in \Theta$ and $\mathrm{b} \in \Theta$. During training, the parameter set is adapted so that the DNN produces the desired input-output behavior as specified by the training data. In the following, we denote the ground-truth target output as y and the output delivered by the network as $\hat{\mathrm{y}}$. For example, one has $\hat{\mathrm{y}} = \mathrm{h}$ for the simple, one-layer network presented above.
The parameters Θ need to be suitably initialized and can then be iteratively optimized by *gradient*

descent [198]. For the optimization, one needs a cost function (often called loss function) $\mathcal{L}$ that measures the deviation between the network output $\hat{\mathrm{y}}$ and the target output y. A popular choice is the cross-entropy:

$$\mathcal{L} = \frac{1}{D} \sum_{d=1}^{D} \left(\mathrm{y}_d \log \hat{\mathrm{y}}_d + (1 - \mathrm{y}_d) \log(1 - \hat{\mathrm{y}}_d) \right). \tag{3.12}$$

From this, the gradient $\mathcal{G}$ of the cost function with respect to the network parameters Θ needs to be determined. Then, the update of the network parameters is given by

$$\Theta \leftarrow \Theta - \mu \cdot \mathcal{G}. \tag{3.13}$$

The meta-parameter μ, a small positive constant, is called learning rate. As with the NMF-based ADT methods, the parameter updates are iterated with index $\ell \in [0 : L]$, with $L = L^{\mathrm{RNN}}$.
In contrast to our simplified example, DNNs are usually a cascade of many layers with individually trainable weights and biases. Although this seems to complicate the derivation of the gradient $\mathcal{G}$, the layered architecture of DNNs allows to use the *backpropagation* algorithm [198] to efficiently calculate gradients for the parameters. In practice, this is usually achieved by using automatic differentiation libraries (e.g. Theano, TensorFlow, etc.).
There are different approaches to utilize training data in this process: using the full dataset (Batch Gradient Descent, BGD), a single data point (Stochastic Gradient Descent, SGD), or a small portion of data points (Mini-Batch Gradient Descent, MBGD) for one update. To accelerate the convergence of gradient descent and to avoid getting stuck in local minima, several modifications have been proposed. Momentum approaches use past update values of the gradient to speed up convergence in problematic areas of the loss function $\mathcal{L}$ (e.g., SGD with momentum [198] and Nesterov accelerated gradient [150]). Adaptive learning rate methods adjust the parameter μ according to the history of past gradients (e.g., Adagrad [55], Adadelta [228], RMSprop [202], and Adam [121]).

3.3.2 Basic RNN Model

In the following sections, four RNN-based ADT systems proposed in the literature [189, 190, 212, 213] will be discussed in detail. Their differences with respect to network configuration, cell architecture, and training strategy will be explained in the corresponding subsections.
RNNs represent an extension of DNNs featuring additional recurrent connections within each layer. The recurrent connections provide the single layers with the previous time step's outputs as additional inputs. The diagram of Figure 3.3b visualizes this concept by a feedback connection from a neuron's output to its input. The equation for the output of an RNN layer at time step

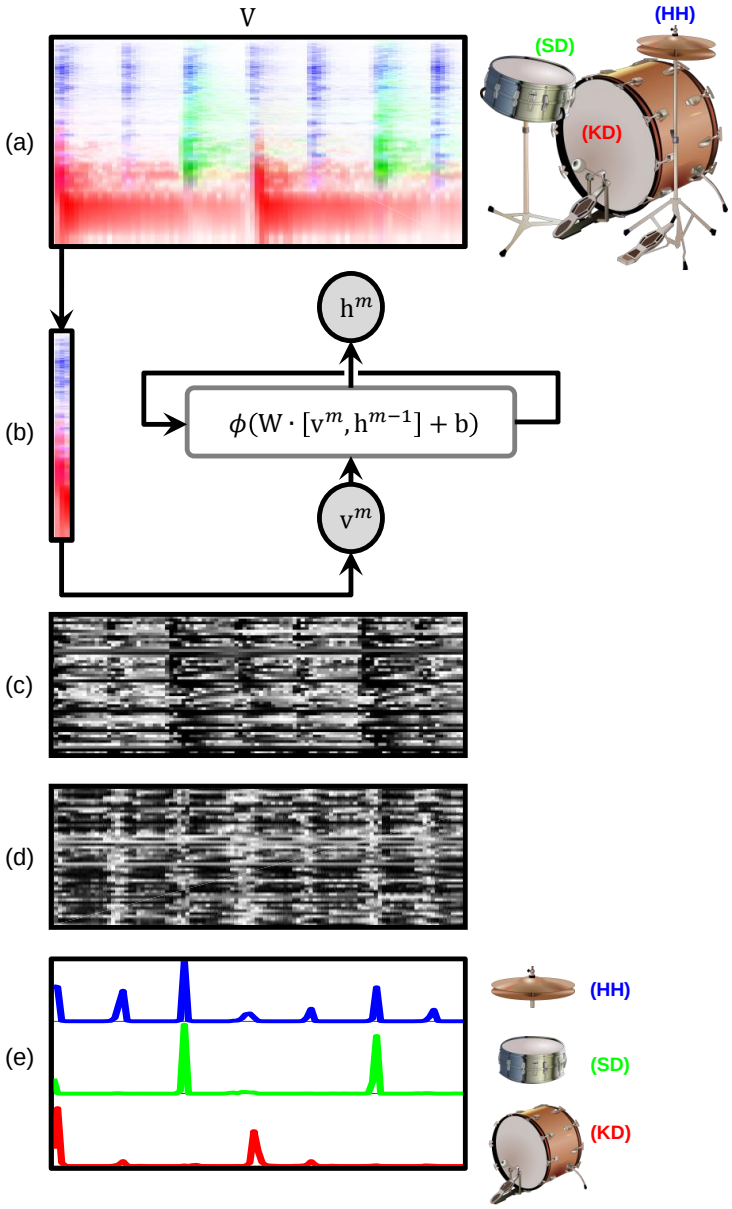

Figure 3.3. Illustration of an RNN-based ADT system. **(a)** Spectrogram of the drum mixture. **(b)** Spectrogram frames are sequentially used as input features for a pre-trained RNN. **(c)** Activations of the first hidden layer. **(d)** Activations of the second hidden layer. **(e)** Activations of the output layer.

m is given by

$$\mathrm{h}^m = \phi\left(\mathrm{W} \cdot \left[\mathrm{v}^m, \mathrm{h}^{m-1}\right] + \mathrm{b}\right), \tag{3.14}$$

where $[:,:]$ denotes concatenation of vector elements. Furthermore, W and b represent the appropriately sized weight matrix and biases vector, while v^m is the current input to the layer and h^{m-1} is the output from the previous time step of the same layer. In case of RNNs with several hidden layers, the output h^m is interpreted as input to the next hidden layer. The feedback of the outputs within the hidden layer acts as a simple form of memory and makes

RNNs suitable for dealing with time series such as the sequence of spectral frames v^m in our spectrogram V.
An algorithm called *backpropagation through time* (BPTT) [217] is utilized to train RNNs, during which the network is thought of being unfolded in time for the length of the time series sequence. Unfolded RNNs become very deep networks, depending on the sequence length used for training. Since deep networks are harder to train, often only subsequences of the time series data are used for training.
In Figure 3.3c and Figure 3.3d, we show the hidden layer activations in a trained RNN. Darker shades of gray encode higher absolute activation. On closer inspection, some structure is visible as the activations tend to be stronger when drum sound events occur in the input. Finally, Figure 3.3e displays the output activations according to our example drum recording. The output activations nicely indicate the onset times of drum sound events. For our example signal, the RNN-based activations are even more pronounced and spiky than the ones obtained via NMFD (cf. Figure 3.2).
For the evaluation in Section 3.4, we use a simple baseline RNN, similar to the plain RNNs in [189, 212]. We compare them against RNNs with more specialized cells, such as Long Short-Term Memory (LSTM) [108] and Gated Recurrent Units (GRU) [31]. In addition to recurrent connections, LSTM cells feature an internal memory, which allows the network to learn long-term dependencies. As an alternative, GRUs can be seen as a modification of standard LSTMs having a smaller number of learnable parameters. The in-depth description of these specialized cells and network architectures is beyond the scope of this thesis. The corresponding publications and important meta-parameters used in our experiments are given in Table 3.1, further details can be found in [223].

3.4 Evaluation

In this section, we provide the details of the evaluation we conducted with the state-of-the-art ADT systems introduced in the last two sections. Specifically, we implemented ten systems from publications within the last five years (cf. Table 2.2) in order to assess and compare their capabilities in a unified experimental framework. The selected algorithms are listed in Table 3.1, where we refer the reader to the original papers. The source code of the implemented systems can be found online.[12,13,14]

[12] `https://github.com/cwu307/NmfDrumToolbox`, last accessed June 14, 2018
[13] `https://github.com/CarlSouthall/ADTLib`, last accessed June 14, 2018
[14] `https://github.com/richard-vogl/dt_demo`, last accessed June 14, 2018

Type	Abbrev.	Reference	Parameters
NMF-based	SANMF	Dittmar and Gärtner [37]	$R = 3, L^{\mathrm{NMF}} = 30, \beta = 4$
	NMFD	Lindsay-Smith et al. [133] Röbel et al. [174]	$R = 3, L^{\mathrm{NMF}} = 30, T = 10$
	PFNMF	Wu and Lerch [220]	$R_{\mathrm{D}} = 3, R_{\mathrm{H}} = 10$ (DTD), $R_{\mathrm{H}} = 50$ (DTP & DTM), $L^{\mathrm{NMF}} = 20$
	AM1	Wu and Lerch [220]	$R_{\mathrm{D}} = 3, R_{\mathrm{H}} = 10$ (DTD), $R_{\mathrm{H}} = 50$ (DTP & DTM), $L^{\mathrm{NMF}} = 20$
	AM2	Wu and Lerch [220]	$R_{\mathrm{D}} = 3, R_{\mathrm{H}} = 10$ (DTD), $R_{\mathrm{H}} = 50$ (DTP & DTM), $L^{\mathrm{NMF}} = 20$
RNN-based	RNN	Vogl et al. [212] Southall et al. [189]	1 hidden layer, $D = 200$, tanh, RMSprop with initial $\mu = 0.005$, mini-batch size = 8 sequences of length 100, weight init uniform ± 0.01, sigmoid outputs, bias init 0
	tanhB	Southall et al. [189]	2 hidden layers, $D = 50$, tanh, Adam with initial $\mu = 0.05$, mini-batch size = 10 sequences of length 100, weight init uniform ± 1, dropout rate 0.25, softmax outputs, bias init 0,
	ReLUts	Vogl et al. [212]	1 hidden layer, $D = 100$, ReLU, RMSprop with initial $\mu = 0.001$, mini-batch size = 8 sequences of length 100, weight init uniform ± 0.01, dropout rate 0.2, sigmoid outputs, bias init 0
	lstmpB	Southall et al. [190]	2 hidden layers, $D = 50$, BLSTMP, Adam with initial $\mu = 0.05$, mini-batch size = 10 sequences of length 100, weight init uniform ± 1, dropout rate 0.25, softmax outputs, bias init 0
	GRUts	Vogl et al. [213]	2 hidden layers, $D = 50$, GRU, RMSprop with initial $\mu = 0.007$, mini-batch size = 8 sequences of length 100, weight init uniform ± 0.1, dropout rate 0.3, sigmoid outputs, bias init 0

Table 3.1. Overview of all implemented systems included in our evaluation

3.4.1 Evaluation Datasets

As indicated earlier, we used two publicly available corpora of drum recordings for our experiments. We processed and partitioned these corpora in such a way that they directly correspond to the three most relevant ADT tasks introduced in Section 2.3. In particular, these are Drum Transcription of Drum-only recordings (DTD), Drum Transcription in the presence of Percussion (DTP), and Drum Transcription in the presence of Melodic instruments (DTM). Table 3.2 gives an overview of the content of these datasets; additional information is provided in the following paragraphs.

D-DTD: This dataset is intended to evaluate DTD performance, i. e., transcription of recordings containing only the three drum instruments KD, SD, HH. A real-world application scenario for this task would be the transcription of single track drum recordings in a studio. This dataset uses the latest version of the IDMT-SMT-Drums corpus [37], which comprises solely drum recordings containing the above-mentioned drum instruments. Each item in the dataset has a ground-truth transcription and comes with training audio files, which contain the used drum sounds in isolation.

D-DTP: This dataset is intended to assess DTP peformance, i. e., transcription of recordings containing other percussion instruments in addition to the drum instruments under observation. To assemble the dataset, we use all items contained in the ENST-Drums minus-one corpus [90], which comprise recordings of full drum kits, including instruments such as CC, RC, HT, MT,

Dataset	Reference	#Onsets Type	Total #items	Avg. Dur.	Subset 1 Origin (#items)	Subset 2 Origin (#items)	Subset 3 Origin (#items)
D-DTD	IDMT-SMT-Drums [37]	8722 KD (2309) SD (1658) HH (4755)	104	15 s	D-DTD-1 RealDrum (20)	D-DTD-2 TechnoDrum (14)	D-DTD-3 WaveDrum (70)
D-DTP	ENST-Drums minus-one [90]	22391 KD (6451) SD (6722) HH (9218)	64	55 s	D-DTP-1 Drummer1 (21)	D-DTP-2 Drummer2 (22)	D-DTP-3 Drummer3 (21)
D-DTM	ENST-Drums accompanied [90]	22391 KD (6451) SD (6722) HH (9218)	64	55 s	D-DTM-1 Drummer1 (21)	D-DTM-2 Drummer2 (22)	D-DTM-3 Drummer3 (21)

Table 3.2. Overview of the three datasets used for our evaluation.

and LT (see Figure 2.1). Again, each item in the dataset has a corresponding ground-truth transcription available. In order to use this information for DTP evaluation, we only consider the annotations for KD, SD, and HH for our performance metrics (see Section 3.4.3). In contrast to D-DTD, this set does not have training audio of isolated drum sound events for each audio item, but only for the three different drum kits that have been used in the sessions. More detailed information about the content of this dataset is provided in the second row of Table 3.2.

D-DTM: This set is intended to evaluate DTM performance, i. e., transcription of polyphonic music recordings containing a variety of melodic instruments in addition to the drum instruments under observation. Again, we use all items contained in the ENST-Drums minus-one dataset, but we create new mixtures using the drum recordings and the corresponding accompaniment recordings. The accompaniments contained in ENST-Drums are partly played on real instruments (e.g., bass, guitar, saxophone, clarinet) and partly on synthesizers. All are temporally aligned to the drum recordings, since the drummers were asked to play along to the backing tracks. We combined accompaniment and drum tracks using a mixing ratio of 1 : 2. We can readily re-use the ground-truth transcriptions of D-DTP since the underlying drum recordings stay the same. We again focus on KD, SD, HH and interpret the melodic accompaniment and the additional percussion as interference making the DTM task the most challenging in our performance comparison.

As shown in the three rightmost columns of Table 3.2, each of the datasets can be naturally split into three subsets. For the IDMT-SMT-Drums corpus, the subsets correspond to the different origins of the drum recordings, namely acoustic drum kits (RealDrum), drum computers (TechnoDrum), and drum sampler software (WaveDrum). For the ENST-Drums corpus, the subsets correspond to three different drummers, each one playing an individual acoustic drum kit.

Evaluation Strategy	Training	Validation	Testing
Eval Random	70% D-DTD	15% D-DTD	15% D-DTD
Eval Subset	{D-DTD-2, D-DTD-3}	50% D-DTD-1	50% D-DTD-1
	{D-DTD-1, D-DTD-3}	50% D-DTD-2	50% D-DTD-2
	{D-DTD-1, D-DTD-2}	50% D-DTD-3	50% D-DTD-3
Eval Cross	70% D-DTP	50% D-DTD	50% D-DTD
	70% D-DTM	50% D-DTD	50% D-DTD

Table 3.3. Summary of the three evaluation strategies applied to the dataset D-DTD. The given percentages denote random selection of items contained in the respective dataset or subset. The curly brackets encode the union of the enclosed subsets.

As layed out in Table 3.2, we denote the individual subsets with the respective dataset name, followed by the suffix -1, -2, and -3. As an example, the subset named D-DTP-2 contains all items featuring the second drummer in the ENST-Drums corpus. In the next section, we will explain why these different subsets are important for our evaluation.

3.4.2 Evaluation Strategies

The goal of our evaluation is to compare the attainable ADT performance of NMF-based and RNN-based systems within a common evaluation framework. As explained in Section 3.2, all ADT systems employing NMF-variants require informed initialization of their spectral bases with averaged drum sound spectra. This step is essential and can be interpreted as some sort of training stage. Similarly, all ADT systems employing RNN-variants require a training stage (see Section 3.3), where a large number of input feature vectors and target output vectors are presented to the network to adjust the internal parameters. Moreover, both NMF and RNN belong to the cluster of activation-based ADT methods whose **AF** activations have to undergo an **ES** stage, which we realize via peak picking. As described in Section 3.1.3, the identification of peak candidates also depends on meta-parameters that have to be optimized.
In our evaluation, we follow the established standards used for evaluating machine learning algorithms. First and foremost, this means that we have to partition the entirety of our data into disjoint sets used for training, validation, and testing. The training data is used to optimize the internal parameters of the selected ADT systems, the validation data is used to optimize hyper-parameters, while the test data is used to measure the performance on unseen data.
We pursue three evaluation strategies explained in the following paragraphs. In Table 3.3, we illustrate how the three strategies apply to the dataset D-DTD. The same principle then applies for the remaining two datasets D-DTP and D-DTM, the only difference being that the datasets need to be swapped.

Eval Random: This strategy evaluates the ADT performance within the "closed world" of each dataset D-DTD, D-DTP, and D-DTM individually. In order to maximize the diversity of the data, all items (regardless of the subset partitions) are randomly split into non-overlapping training, validation, and testing sets.

Eval Subset: This strategy also evaluates the ADT performance within each dataset but using a three-fold cross-validation. To this end, all possible combinations arising from two of the subsets are merged and once used as training data. In Table 3.2, we show the resulting training sets enclosed in curly brackets. For each training run, the remaining subset is evenly split into validation and testing set.

Eval Cross: This strategy evaluates the generalization capabilities across different datasets. To this end, each of the datasets is once split into into validation and test data. The remaining two datasets are used as individual training instances. For practical reasons, we re-use the pre-trained models from Eval Random for this evaluation strategy.

3.4.3 Parameters and Performance Metrics

The **FR** considered in our evaluation is computed via STFT with a blocksize of $N = 2048$ and a hopsize of $\frac{N}{4} = 512$. Since all items have a sampling rate of 44.1 kHz, the frequency resolution of the STFT is approx. 21.5 Hz and the temporal resolution is approx. 11.6 ms. As window function, we use a symmetric Hann-window of size N.
We use the F-measure as main performance metric. In Section 3.1.3, we introduced the binary output matrix O representing the frame-wise results of the peak picking stage. Frames that are marked as containing a peak corresponding to the start of a drum sound event are considered to be *true positives* (TP) if they fall within 50 ms of the corresponding ground-truth annotation. Marked frames that do not coincide with an annotated drum sound event are counted as *false positives* (FP). If no marked frames occur within the 50 ms neighborhood of an annotated drum sound event, these instances are counted as *false negatives* (FN). The three quantities define the standard F-measure $F = 2 \cdot \mathrm{TP} / (2 \cdot \mathrm{TP} + \mathrm{FP} + \mathrm{FN})$.

3.4.4 Results and Discussions

In this section, we provide a top-down summary of the main findings to highlight the essence of our evaluation. For the sake of completeness and reproducibility, the table with all detailed

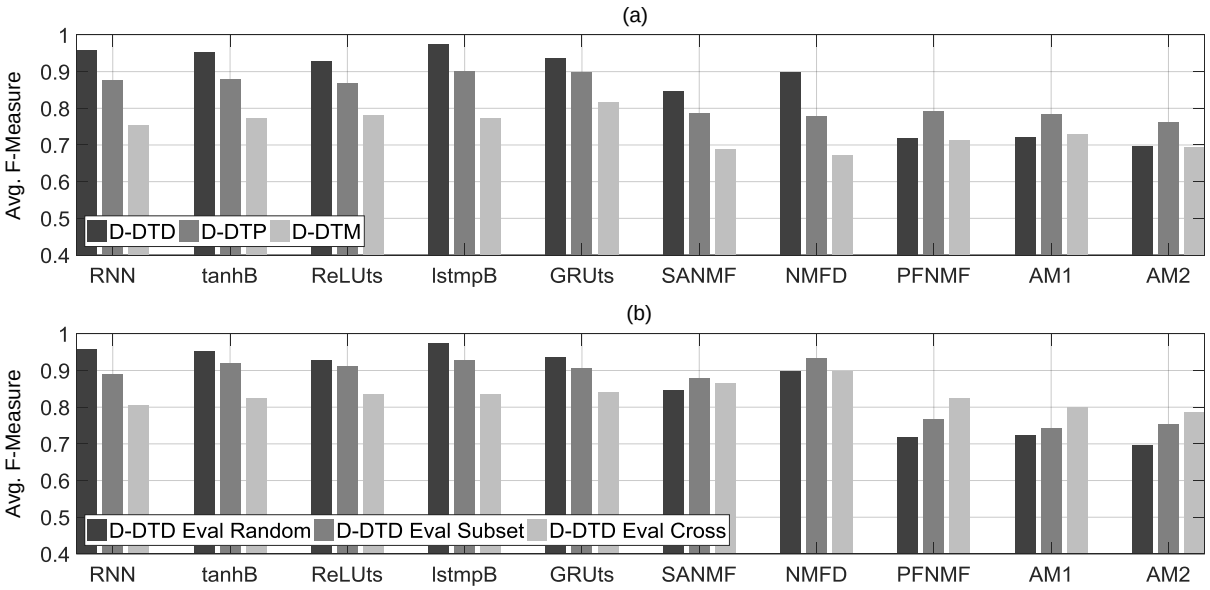

Figure 3.4. Summary of our evaluation. **(a)** F-measure for different ADT systems using the Eval Randomevaluation strategy on our three datasets D-DTD, D-DTP, and D-DTM. **(b)** F-measure for the same ADT systems employed on the D-DTD dataset while switching between the evaluation strategies Eval Random, Eval Subset, and Eval Cross.

experimental results can be found on our complementary website[15].

In Figure 3.4a, we assess how well the selected systems can cope with ADT tasks of increasing complexity. To this end, we show the average F-measure across our three datasets in the evaluation scenario Eval Random. This evaluation scenario provides the most ideal case, in which the training data is likely to be representative for the test data. As expected, the highest results are achieved with the least complex dataset D-DTD. From the family of RNN-based methods, **lstmpB** is the best-performing system with approximately 0.97 F-measure, i.e., almost perfectly solving the DTD task. From the family of NMF-based methods, **NMFD** scores best, but falls short of all RNN-based systems.

For the more challenging dataset D-DTP, the performance of all systems drops, except for **PFNMF** variants. Compared to **SANMF** and **NMFD**, they seem to cope better with additional percussion instruments that are not reflected in the pre-trained NMF templates. Finally, for the most challenging D-DTM dataset, **GRUts** is the only system that surpasses 0.8 F-Measure. Once again, the performance of all other systems deteriorates. Only the **PFNMF**-variants can partly compensate for the performance drop, with **AM1** scoring best among the NMF-methods.

In Figure 3.4b, we assess the generalization capabilities of the evaluated systems. To this end, we stay with the dataset D-DTD and sweep through our evaluation scenarios. This dataset is the least challenging one and lends itself well to studying the impact of training. We observe that the RNN-based systems are quite susceptible to mismatches in the training data. Performing

[15] `http://www.audiolabs-erlangen.de/resources/MIR/2017-DrumTranscription-Survey/`, last accessed June 14, 2018

RNN-training on the Eval Subset data already leads to a slight decrease. The performance drop is even more pronounced when the training is based on the Eval Cross data. In contrast, the NMF-based methods either stay stable or improve their performance through the different training scenarios. This is due to the following, fundamental difference between both algorithms. The internal parameters of RNNs are adapted during training time and kept fixed at test time. In contrast, NMF templates are only initialized based on training data. At test time, they may adapt to the characteristics of the test data.
It should be noted that we present here the averaged results, i.e., the Eval Subset training results are averaged over the test splits of D-DTD-1, D-DTD-2, and D-DTD-3. Likewise, the Eval Cross training results are averaged over training with D-DTP and D-DTM.
Based on the above results, the following trends can be concluded: First, RNN-based systems generally outperform NMF-based systems. Even the basic RNN system (included as a baseline) performs on a par with the other systems in most cases. Since RNNs exploit the temporal dependencies in the input data, they have the potential to learn the underlying structure and temporal context. However, for less challenging data, NMF-based systems may provide competitive results at a much lower computational load for training. Second, the margin between the strongest and weakest systems decreases as the ADT task gets more challenging. This result indicates the typical vulnerability against the interference of other instruments that is common for all state-of-the-art systems. Third, different training strategies have less impact on NMF-based systems, whereas the performance drop from Eval Random over Eval Subset to Eval Cross is noticeable for RNN-based systems. Since Eval Random offers more diversity (i.e., more training examples similar to the ones in the test set), it is expected to be more advantageous for RNNs. On the contrary, when the test data contains unseen examples, RNNs become less reliable.

3.5 Conclusions and Further Notes

In this chapter, we provided an introduction to state-of-the-art ADT approaches that are based on variants of NMF and RNN. Furthermore, we conducted a systematic performance comparison of these two families of algorithms under well-controlled experimental conditions. From our experiments, one may conclude that RNN-based methods seem to be most promising. They are the recommended choice when large and diverse training data with high-quality annotations is available. NMF-based methods, on the other hand, provide decent performance with only little training data required. If the training data requirements mentioned above can be fulfilled, the task of Drum Transcription from Drum-only Recordings (DTD) can be considered as practically solved by state-of-the-art systems.
The more difficult tasks of drum transcription in the presence of additional percussion instruments (DTP) and/or melodic instruments (DTM) still pose a major challenge to the current ADT

techniques. This becomes evident when taking a look at the outcomes of the recently conducted MIReX competition on ADT[16]. Surprisingly, one can observe that the results of the best current systems are only slightly above the winning method by Yoshii et al. [227] from more than 12 years ago[17]. Notably, Yoshii et al. neither used RNN nor NMF, but their own meticulously crafted and optimized spectrogram template adaption method called *AdaMa*. However, they did care about pre-processing steps to attenuate the influence of pitched instruments in the music recordings.

For the following chapters of this thesis, we are going to focus on the NMF-based methods. Although they do not score as well as RNN-based methods for ADT purposes, they have several advantages for our purpose. First, they do not require large training datasets. Second, they are reasonably light-weight in terms of computational requirements during training. Third, they can adapt to new signal characteristics not reflected in the training data. And fourth, the transition from automatic drum transcription to drum sound separation is relatively straight-forward, as we will see in the following chapter.

[16] `http://www.music-ir.org/mirex/wiki/2017:Drum_Transcription_Results`, last accessed June 14, 2018
[17] `http://www.music-ir.org/mirex/wiki/2005:Audio_Drum_Detection_Results`, last accessed June 14, 2018

Part II

Drum Sound Separation

Chapter 4

Score-Informed Separation of Drum Recordings

The work in this chapter is mainly based on our contribution in [39].

This chapter addresses the extraction of high-quality component signals from drum solo recordings (breakbeats) for music production and remixing purposes. Specifically, we employ audio source separation techniques to recover sound events from the drum sound mixture corresponding to the individual drum strokes. Our separation approach is based on an informed variant of Non-Negative Matrix Factor Deconvolution (NMFD) that has been proposed and applied to drum transcription and separation in earlier works. In this chapter, we systematically study the suitability of NMFD and the impact of audio- and score-based side information in the context of drum separation. In the case of imperfect decompositions, we observe different cross-talk artifacts appearing during the attack and the decay segment of the extracted drum sounds. Based on these findings, we propose and evaluate two extensions to the core technique. The first extension is based on applying a cascaded NMFD decomposition while retaining selected side information. The second extension is a time-frequency selective restoration approach using a dictionary of single note drum sounds. For all our experiments, we use a publicly available dataset consisting of multi-track drum recordings and corresponding annotations that allows us to evaluate the source separation quality. Using this test set, we show that our proposed methods improve the quality of the component signals.

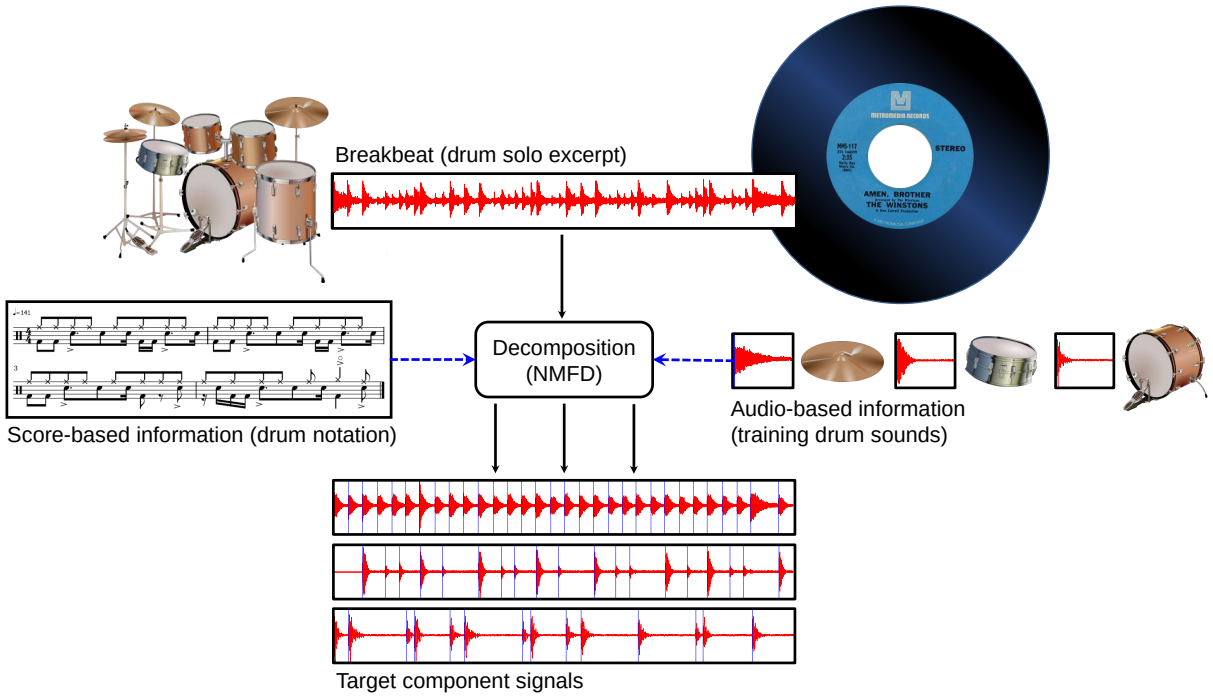

Figure 4.1. Illustration of our application scenario, the decomposition of a recording's drum solo section (i.e., breakbeat). The the targeted output are the separated single tracks with annotated onset times of all drum events. Score-based and audio-based side information may guide the decomposition, as indicated by the dashed blue arrows.

4.1 Introduction

The term "breakbeat" typically describes drum solo excerpts taken from funk, soul and jazz records of the 1960s to 1980s. Breakbeats often provide the rhythmic foundation of many songs in modern, sample-based music genres such as hip-hop, drum'n'bass, and big beat [1]. One of the most famous examples, the "Amen Break", stems from the song "Amen, Brother", which was recorded in 1969 by the funk and soul band "The Winstons" (the original vinyl record is shown in Figure 4.1). After 86 seconds into the song, drummer G. C. Coleman plays a 4-bar drum solo in 4/4 time at about 140 BPM. With the advent of affordable sampling technology, this particular breakbeat (shown by the waveform in the upper part of Figure 4.1) later became extremely popular and was extensively re-used [41, 109, 110]. In [163, pp. 318–328] other iconic breakbeats, such as the "Funky Drummer" and "Apache" are discussed. The popularity of breakbeats results from their rhythmic intricacies in conjunction with a distinct sound that can be attributed to technical limitations of the era's recording equipment, as well as unique acoustic conditions in the recording environment.

The typical drum instruments encountered in breakbeats can be roughly divided in two classes: membranophones and idiophones. Membranophones usually occupy the lower frequency regions of the spectrum. Examples ordered according to their dominant frequency regions are the kick

drum, tom tom, snare drum, timbales, conga, and bongo. Continuing with idiophones, we have cowbells, woodblocks, shaker, cymbals, hi-hat, and tambourine. The rhythmic gist of breakbeats is typically dominated by kick drum, snare drum, and hi-hat (or ride cymbals). Crash cymbals and percussion instruments may be used in addition to create more variation in the rhythmic patterns. These instruments rarely exhibit a clear pitch and are often strongly overlapping in frequency due to their broad band spectral characteristics. Moreover, a single drum or percussion instrument can produce quite a variety of sounds, depending on the playing technique. In this chapter, we focus on kick drum, snare drum, and hi-hat. However, similar techniques may be used for other drum instruments.

In the early days of sampling, entire breakbeat sections were simply used as loops providing the rhythmic foundation for new songs. Later on, thanks to more powerful music editing equipment and software, it became common practice to segment these breakbeats according to the underlying rhythmical grid and swap or repeat certain sections to create entirely new rhythms. This technological and artistic development is illustrated in great detail in [109, pp. 77–132]. As a typical example, the drum transcription of a track by drum'n'bass artist "Shy FX" — who uses an entirely rearranged, rhythmically complex version of the "Amen Break" — is given in [163, pp. 327]. Other artists like "Squarepusher", "Photek", or "Venetian Snares" pushed the boundaries of breakbeat programming, by combining resequenced rhythms with heavy use of audio processing techniques such as resampling, filtering, or time-scale modification in a virtuosic manner. However, even in these extreme forms, the breakbeats are still based on rhythmically segmented excerpts of the original mixture, i.e., the original drum sound events still occur simultaneously in a monolithic mixture.

4.1.1 Goals and Challenges

In modern music production, drums are typically recorded using multiple microphones for multi-track processing of the individual drum instruments. Such multi-tracks are the basis for drum leakage suppression [122] and drum replacement [37]. Since breakbeats originate from vinyl recordings of the 1960s to 1980s, we only have the mixture of all drums available since it is uncommon to find multi-tracks. Aiming for intelligent applications for music production and remixing, our goal is to decompose those breakbeats into their constituent drum sounds.

Although research on source separation techniques for signal decomposition has made significant progress over the last years [67], there are still open problems. Most importantly, blind source separation is considered to be particularly challenging due to the inherent ambiguities (spectral overlap, temporal interdependence). Thus, specialized techniques such as harmonic-percussive source separation (HPSS) [199] aim at splitting a music recording into harmonic (e.g., melodic instruments, tonal components) and percussive (e.g., drums and percussion, transients) compo-

nents. Moreover, drum separation is usually realized as a two-stage process, by first deriving an approximate transcription and then performing the separation. Although Non-Negative Matrix Factor Deconvolution (NMFD) [133, 174, 185] has proven to be suitable for both aspects, we focus in this chapter on the second stage. Therefore, we assume that a transcription is already available and concentrate on obtaining a high-fidelity separation of drum sound events. Given additional side information (as shown in Figure 4.1), we need to investigate the benefit of imposing different constraints to NMFD [67, 153, 210]. Since imperfect decomposition may result in audible cross-talk (leakage, interference) between the individual drum instruments, we focus on improving their perceptual quality through signal-specific restoration [28, 42].

4.1.2 Contribution

The main contributions of this chapter are as follows. First, we adapt and discuss an informed variant of NMFD for separating drum mixtures. Our approach is inspired by the original work [185] that outlines the specific application of NMFD to drum transcription and separation. Using a publicly available dataset (see Section 4.1.3), we systematically evaluate the benefit of imposing score-informed and audio-informed constraints to NMFD (see Section 4.3). To our knowledge, this chapter is the first that specifically investigates these influential factors in the context of drum separation. In our analysis, we identify two types of unwanted cross-talk artifacts that can occur due to an imperfect NMFD decomposition. The main technical contributions of our chapter consist of two novel post-processing techniques to improve the initial NMFD decomposition (see Section 4.4). The first idea is a cascaded application of NMFD on the initial separations to reduce cross-talk. As a second idea, we propose a dictionary-based method (extending [42]) to restore the attack sections of the separated drums. Finally, we demonstrate the applicability of the proposed method for real-world breakbeat recordings, discuss other applications, and point out open issues for future work (see Section 4.5). Further relevant work by other authors will be discussed in the appropriate sections.

4.1.3 Data Sources

In addition to the technical challenges discussed in Section 4.1.1, we face the problem that there is no best-practice procedure for evaluating the quality of breakbeat separation, since no ground truth data is available. Therefore, we use the "IDMT-SMT-Drums" dataset[18], a publicly available corpus of drum solo recordings in order to mimic real-world breakbeats. This corpus comprises only drum recordings and is enriched with side information such as the true "oracle" component signals and their precise onset positions and instrument types. In Figure 4.1, we

[18] `http://www.idmt.fraunhofer.de/en/business_units/smt/drums.html`, last accessed June 14, 2018

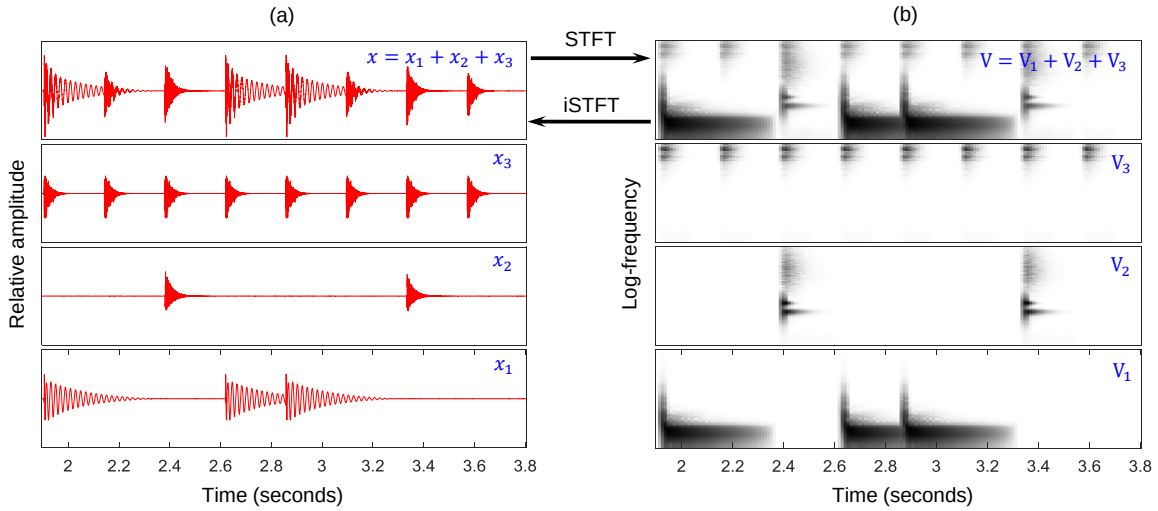

Figure 4.2. Illustration of our signal model. **(a)** Mixture signal x is the sum of $C = 3$ component signals x_c (x_1: kick drum, x_2: snare drum, x_3: hi-hat). **(b)** Time-frequency representation of the mixture's magnitude spectrogram V and $C = 3$ component magnitude spectrograms V_c. For better visibility, we use a logarithmic frequency axis and logarithmic magnitude.

provide an overview of the available data sources. At the top, we start with the breakbeat waveform taken from an original record. At the bottom, we have the targeted component signals, overlaid with the discrete onset times (vertical blue lines). To the left, we have score-based side information (onset times, instrument types). To the right, we have audio-based side information (training drum sounds).

In addition to "IDMT-SMT-Drums", we use a second dataset of isolated drum sounds from commercial sampling CDs. It is completely disjoint from the "IDMT-SMT-Drums" dataset, i.e., the drum sounds originate from different sources. It contains 201 kick drum, 304 snare drum, and 188 hi-hat sounds, amounting to approximately four minutes total running time. This second audio dataset is used for parameter optimization (see Section 4.3.3) and component restoration (see Section 4.4.2).

4.2 Baseline Decomposition

In this section, we first introduce the notation and signal model used throughout this chapter. Second, we review the NMFD decomposition method as proposed in [185]. Subsequently, we describe our strategy to enforce score-based and audio-based constraints on NMFD, before we finally provide the details about the signal reconstruction from the NMFD results.

4.2.1 Notation and Signal Model

We consider the real-valued, discrete time-domain signal $x : \mathbb{Z} \to \mathbb{R}$ to be a linear mixture $x := \sum_{c=1}^{C} x_c$ of $C \in \mathbb{N}$ component signals x_c. As shown in Figure 4.2a, each component signal contains drum sound events corresponding to individual drums instruments. The mixture signal in Figure 4.2 (synthetic drum sounds of a Roland TR-808 drum machine) will serve as a running example for illustration purposes.
As detailed in Section 4.2.2, we decompose a time-frequency (TF) representation of x as depicted in Figure 4.2b. To this end, we employ the Short-Time Fourier Transform (STFT) as follows. Let $\mathcal{X}(m,k)$ be a complex-valued TF coefficient at the m^{th} time frame for $m \in [1:M]$ and k^{th} spectral bin for $k \in [0:K]$. The coefficient is computed via frame-wise Discrete Fourier Transform (DFT) as

$$\mathcal{X}(m,k) := \sum_{n=0}^{N-1} x(n+mH)w(n)\exp(-2\pi ikn/N), \tag{4.1}$$

where $w : [0:N-1] \to \mathbb{R}$ is a suitable window function of even blocksize $N \in \mathbb{N}$, and $H \in \mathbb{N}$ is the hop size parameter. The number of frequency bins[19] is $K = N/2$ and the number of spectral frames $M \in \mathbb{N}$ is determined by the number of signal samples available. From $\mathcal{X}$, the magnitude spectrogram $\mathcal{A}$ and the phase spectrogram φ are derived as

$$\mathcal{A}(m,k) := |\mathcal{X}(m,k)|\,, \tag{4.2}$$

$$\varphi(m,k) := \angle\mathcal{X}(m,k), \tag{4.3}$$

with $\varphi(m,k) \in [0, 2\pi)$.

4.2.2 Spectrogram Decomposition via NMFD

In this section, we briefly review the NMFD method that we employ for decomposing the TF-representation of x. Previous works [133, 174, 185] successfully applied NMFD, a convolutive version of NMF, for drum transcription and separation. Alternatively, Probabilistic Latent Component Analysis (PLCA) [184] and its convolutive (or shift-invariant) extensions [186] have been used for music transcription purposes [12].
In this chapter we follow the NMFD perspective, whose convolutive model assumes that all audio events in one of the component signals can be explained by a prototype event that acts as an impulse response to some impulsive activation (e.g., striking a particular drum). In Figure 4.2b,

[19] Since x is real-valued, its corresponding spectrum is Hermitian. Thus, we assume that the redundant frequency bins in the upper half of the spectrum are discarded.

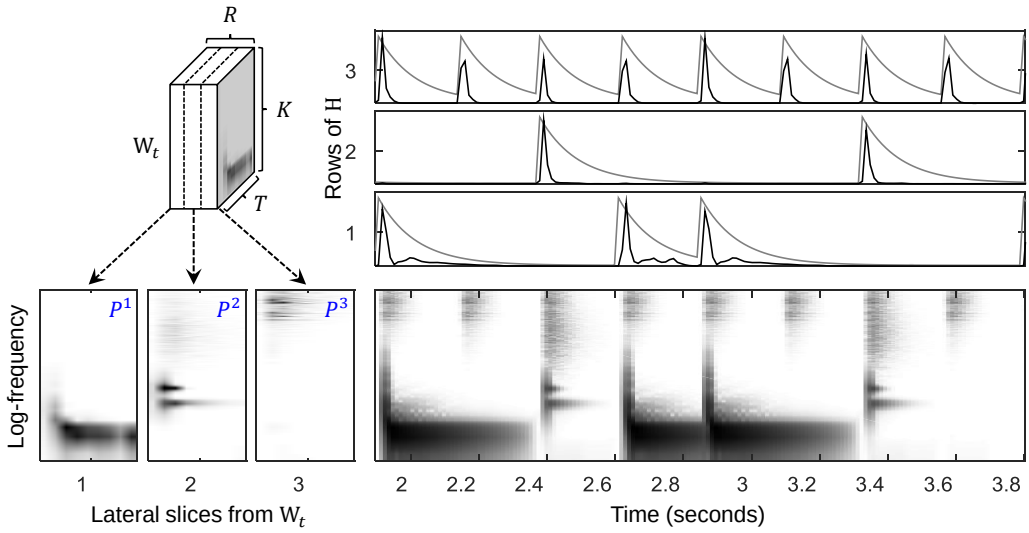

Figure 4.3. NMFD templates and activations computed for the example drum recording from Figure 4.2. The magnitude spectrogram V is shown in the lower right plot. The leftmost panels show the spectrogram templates P^r for $r \in [1:R]$ and $R = 3$, which have been obtained via NMFD. Their corresponding activations in H are shown as black curves in the three top plots. The gray curves show the score-informed initialization $\mathrm{H}^{(0)}$.

one can see this kind of behavior in the hi-hat component V_3: all eight instances of the hi-hat drum sound event are almost identical copies.

Let $\mathrm{V} := \mathcal{A}^\top \in \mathbb{R}_{\geq 0}^{K \times M}$ be a non-negative matrix representing a transposed version of the mixture's magnitude spectrogram $\mathcal{A}$. Our objective is to decompose V into component magnitude spectrograms V_c that correspond to the distinct drum instruments as shown in Figure 4.2b. Conventional NMF can be used to compute a factorization $\mathrm{V} \approx \mathrm{W} \cdot \mathrm{H}$, where the columns of $\mathrm{W} \in \mathbb{R}_{\geq 0}^{K \times R}$ represent spectral basis functions (also called templates) and the rows of $\mathrm{H} \in \mathbb{R}_{\geq 0}^{R \times M}$ contain time-varying gains (also called activations). The rank $R \in \mathbb{N}$ of the approximation (i.e., number of components) is an important but generally unknown parameter. In Section 4.2.3, we will see that it can usually be set to the number of unique instruments C.

NMFD extends the NMF model to the convolutive case by using two-dimensional templates so that each of the R spectral bases can be interpreted as a magnitude spectrogram snippet consisting of $T \ll M$ spectral frames. The convolutive spectrogram approximation $\mathrm{V} \approx \tilde{\mathrm{V}}$ is modeled as

$$\tilde{\mathrm{V}} := \sum_{t=0}^{T-1} \mathrm{W}_t \cdot \overset{t\rightarrow}{\mathrm{H}}, \tag{4.4}$$

where $\overset{t\rightarrow}{(\cdot)}$ denotes a frame shift operator. As before, each column in $\mathrm{W}_t \in \mathbb{R}_{\geq 0}^{K \times R}$ represents the spectral basis of a particular component, but this time we have T different versions W_t, with $t \in [0 : T-1]$ available. If we take lateral slices along the columns of W_t, we can obtain R prototype magnitude spectrograms $\mathrm{P}^r \in \mathbb{R}_{\geq 0}^{K \times T}$ as shown in the $R = 3$ plots in the lower left of Figure 4.3. NMFD typically starts with a suitable initialization (with random values or

constant values) of matrices $\mathrm{W}_t^{(0)}$ and $\mathrm{H}^{(0)}$. Subsequently, these matrices are iteratively updated to minimize a suitable distance measure between the convolutive approximation $\tilde{\mathrm{V}}$ and V. In this work, we use the update rules detailed in [185], which extend the well-known update rules for minimizing the Kullback-Leibler (KL) Divergence [130] to the convolutive case. The modified update rules are given by

$$\mathrm{W}_t^{(\ell+1)} = \mathrm{W}_t^{(\ell)} \odot \frac{\frac{\mathrm{V}}{\tilde{\mathrm{V}}^{(\ell)}} \cdot \begin{pmatrix} \overset{t\rightarrow}{\mathrm{H}}^{(\ell)} \end{pmatrix}^\top}{\mathrm{J} \cdot \begin{pmatrix} \overset{t\rightarrow}{\mathrm{H}}^{(\ell)} \end{pmatrix}^\top} \tag{4.5}$$

$$\mathrm{H}^{(\ell+1)} = \mathrm{H}^{(\ell)} \odot \frac{\left(\mathrm{W}_t^{(\ell+1)}\right)^\top \cdot \left[\frac{\mathrm{V}}{\tilde{\mathrm{V}}^{(\ell)}}\right]^{\leftarrow t}}{\left(\mathrm{W}_t^{(\ell+1)}\right)^\top \cdot \mathrm{J}} \tag{4.6}$$

for $t = 0, 1, 2, \ldots, T-1$ and $\ell = 0, 1, 2, \ldots, L^{\mathrm{NMF}}$ for some $L^{\mathrm{NMF}} \in \mathbb{N}$. The symbol $\odot$ denotes element-wise multiplication and the division is also understood element-wise. Furthermore, $\mathrm{J} \in \mathbb{R}^{K \times M}$ denotes an all-one matrix.

4.2.3 Informed NMFD

When striving for good perceptual quality of the separated target signals, many authors propose to use priors for the decomposition [33, 51, 67, 153, 168, 210]. This has the advantage that the separation can be guided and constrained by information on the approximate location of component signals in time (onset, offset) and frequency (pitch, timbre). Proper initialization of $\mathrm{W}_t^{(0)}$ and $\mathrm{H}^{(0)}$ has proven to be an effective means to enforce convergence of NMF as well as NMFD to the desired, musically meaningful solution.

One possibility is to impose score-based constraints derived from a time-aligned, symbolic transcription [66]. For the moment, let us assume that we have such side information specifying the onset time and drum instrument type for each of the audio events. From that transcription, we derive the number of unique instruments C and set $R = C$.

Furthermore, we initialize the rows of $\mathrm{H}^{(0)}$ as follows: each frame corresponding to an onset of the respective drum instrument is initialized with an impulse of unit amplitude, all remaining frames with a small, positive constant. Afterward, we apply a nonlinear exponential moving average filter with a decay parameter $0 \leq \lambda \leq 1$ to this impulse sequence. The typical outcome of this initialization is shown in the top three plots of Figure 4.3 (gray curves). Using this kind of initialization, a few NMFD iterations are sufficient to learn spectrogram templates corresponding to the prototype spectrograms of the individual drum instruments. The learned activation functions then amount to the deconvolved activation of all occurrences of that particular drum

instrument throughout the recording. A typical decomposition result is shown in Figure 4.3, where one can see that the extracted templates (three leftmost plots) indeed resemble prototype versions of the onset events in V (lower right plot). Furthermore, the location of the score-informed impulses in the extracted H (three topmost plots) are very close to the local maxima of the score-informed initialization (gray curves).

In [14, 66, 168], best results were obtained by initializing both activations and templates using prior information. For separation of pitched instruments (e.g., piano), prototypical overtone series can easily be constructed to initialize certain TF-bins in $\mathrm{W}_t^{(0)}$. As the exact frequencies are unknown, a neighborhood around these positions can then be initialized with non-zero values in the templates, while setting the remaining entries to zero, see [66, 168] for details.

For drums, it is more difficult to model prototype templates. Thus, it has been proposed to initialize them with temporally averaged or factorized spectrograms obtained from isolated drum sounds [8, 37, 76, 133, 160, 174]. If we have training examples of the true drum sounds available (see Section 4.3.1), we use the following initialization scheme. For each of our $c \in [1 : C]$ drum instruments, a prototype magnitude spectrogram P^c is learned from the training signals, using a preliminary NMFD with $R = 1$. Subsequently, we concatenate the learned $\mathrm{P}^1, \ldots, \mathrm{P}^c$ as an audio-informed initialization for $\mathrm{W}_t^{(0)}$.

4.2.4 Component Signal Reconstruction

In the following, we describe how to further process the NMFD results to reconstruct the target time-domain component signals. Let $\mathrm{H} \in \mathbb{R}_{\geq 0}^{R \times M}$ be the activation matrix and $\mathrm{W}_t \in \mathbb{R}_{\geq 0}^{K \times R}$ be the spectrogram bases previously learned by NMFD. Then, we define for each $c \in [1 : R]$ the matrix $\mathrm{H}_c \in \mathbb{R}_{\geq 0}^{R \times M}$ by setting all elements to zero except for the c^{th} row that contains the activations previously learned by NMFD. We approximate the c^{th} component magnitude spectrogram by the following modification of (4.4):

$$\tilde{\mathrm{V}}_c := \sum_{t=0}^{T-1} \mathrm{W}_t \cdot \overset{t\rightarrow}{\mathrm{H}_c}. \tag{4.7}$$

Since the NMFD model yields only a low-rank approximation of V, spectral nuances may not be fully captured. This can result in unnatural artifacts in the time-domain reconstruction. In order to remedy this problem, it is common practice to calculate soft masks that can be interpreted as weighting factors reflecting the contribution of $\tilde{\mathrm{V}}_c$ to the mixture V per TF-bin. Intuitively, the magnitude ratio of the desired component to the sum of all components might be considered a good choice. A more generalized view is explained in [134]. According to their work, we derive

the mask corresponding to the c^{th} component as

$$\mathrm{M}_c := \tilde{\mathrm{V}}_c^{\alpha} \oslash \left(\epsilon + \sum_{c=1}^{R} \tilde{\mathrm{V}}_c^{\alpha} \right), \tag{4.8}$$

where $\oslash$ denotes element-wise division, ϵ is a small positive constant to avoid division by zero, and the exponent $0 \leq \alpha \leq 2$ is also applied in an element-wise fashion. We obtain the estimate $\hat{\mathrm{V}}_c \in \mathbb{R}_{\geq 0}^{K \times M}$ of the component magnitude spectrogram by setting

$$\hat{\mathrm{V}}_c := \mathrm{V} \odot \mathrm{M}_c, \tag{4.9}$$

with $\odot$ denoting element-wise multiplication. This procedure is referred to as α-Wiener filtering. Further discussion of an appropriate choice for α is provided in [134]. Intuitively speaking, α acts as a selectivity parameter that can be used to trade off between suppression of unwanted components and separation artifacts.

In order to reconstruct an estimated target component signal $\hat{x}_c$, we set $\hat{\mathcal{X}}_c := \hat{\mathcal{A}}_c \odot \exp(i\hat{\varphi}_c)$, where $\hat{\mathcal{A}}_c = \hat{\mathrm{V}}_c^{\top}$ and $\hat{\varphi}_c$ is an estimate of the component phase spectrogram. It is common practice to use the mixture phase information φ as an estimate for $\hat{\varphi}_c$ and to invert the resulting $\hat{\mathcal{X}}_c$ via the reconstruction method known as LSEE-MSTFT [96]. The method first applies the inverse Discrete Fourier Transform (DFT) to each spectral frame in $\hat{\mathcal{X}}_c$, yielding a set of intermediate time signals y_m, with $m \in [0 : M-1]$, defined by

$$y_m(n) := \frac{1}{N} \sum_{k=0}^{N-1} \hat{\mathcal{X}}_c(m,k) \exp(2\pi i k n / N), \tag{4.10}$$

for $n \in [0 : N-1]$ and $y_m(n) := 0$ for $n \in \mathbb{Z} \setminus [0 : N-1]$. Second, the least squares error reconstruction is achieved by

$$\hat{x}_c(n) := \frac{\sum_{m \in \mathbb{Z}} y_m(n - mH) w(n - mH)}{\sum_{m \in \mathbb{Z}} w(n - mH)^2}, \tag{4.11}$$

$n \in \mathbb{Z}$, where the analysis window w is re-used as a synthesis window. In [38], we show how an additional, iterative phase reconstruction stage using score-based information can improve the reconstruction of the transient signal portions and attenuate the so-called pre-echos. However, the issues inherent to phase reconstruction are out of the scope of this work, where we focus on magnitude spectrograms only.

4.3 Baseline Experiment

In this section, we present the experiments and results obtained by varying the degree of side-information that we provide to the core decomposition algorithm as described in Section 4.2. Our aim is to gain insights about the capabilities and limitations of the approach. Specifically, we want to assess which combination of score-based and audio-based information is most influential to the separation quality.

4.3.1 Dataset and Metrics

As already indicated in Section 4.1.3, we use the publicly available "IDMT-SMT-Drums" dataset[20] for our separation experiments. In the "WaveDrum02" subset, there are 60 drum loops, each given as perfectly isolated single track recordings (i.e., oracle component signals) of the three instruments kick drum (KD), snare drum (SD), and hi-hat (HH). Additionally, the exact onset times of all drum sound events are available per individual drum instrument. This scenario allows us to mimic real-world breakbeats, for which ground-truth multi-track data is hard (if not impossible) to obtain. All 3×60 component signals in the "WaveDrum02" subset are mono files in uncompressed PCM WAV format with 44.1 kHz sampling rate, 16 Bit. The average duration is around 14 s, the shortest drum recording is 8 s, the longest lasts 17 s. By mixing all three single tracks corresponding to each of the test items, we obtain 60 mixture signals (MIX) that serve as input to the NMFD decomposition.
We use the PEASS Toolkit [62] in order to evaluate the quality of our reconstructed component signals. The PEASS Toolkit delivers four objective metrics based on energy ratios, namely the Signal to Distortion Ratio (SDR), Image to Spatial Distortion Ratio (ISR), Signal to Interference Ratio (SIR), and Signal to Artifact Ratio (SAR), as well as four perceptually motivated scores, namely the Overall Perceptual Score (OPS), Target Perceptual Score (TPS), Interference Perceptual Score (IPS), and Artifact Perceptual Score (APS). For the sake of brevity, we focus on the SDR, SIR and IPS metrics in this chapter. We follow the recommendations from [28] and favor perceptual quality of the individual sources over the ability to perfectly reconstruct the original mixture from its components. Thus, we emphasize interference-related metrics that correlate to the amount of cross-talk between the component signals.

4.3.2 Experimental Setup

Intuitively, we expect that the NMFD decomposition will improve with an increasing amount of additional information to guide the path of the parameter updates in (4.5) and (4.6). In our baseline experiment, we use different combinations of the data sources in order to assess the

[20] `http://www.idmt.fraunhofer.de/en/business_units/smt/drums.html`, last accessed June 14, 2018

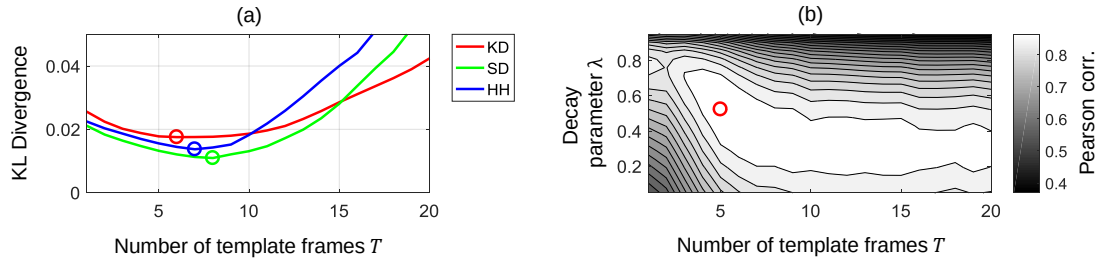

Figure 4.4. Parameter optimization results shown for kick drum (KD), snare drum (SD), and hi-hat (HH), respectively. **(a)** The average KL Divergence per instrument type under varying number of template frames T. **(b)** Outcome of a grid search for the optimal combination of T and decay parameter λ relevant when using score information.

impact of score-based and audio-based information on separation quality. In Table 4.1, we present an overview of the experimental setup in six different test cases. The second column specifies which method delivers the estimate for the approximate component magnitude spectrogram $\tilde{\mathrm{V}}_c$ before α-Wiener filtering. This is almost always the NMFD; only in Case 0 we use the magnitude STFT of the oracle (ground truth) component signals instead. Thus, Case 0 is our expected upper bound for separation quality. The third column specifies how the initial values of the spectrogram templates in $\mathrm{W}_t^{(0)}$ are set. We use initializations obtained from preliminary NMFD applied to perfectly isolated training samples (see Section 4.2.3) of the drum sounds in Case 1a, Case 1b, and Case 2. Furthermore, we fix these initial values in Case 1a, effectively disabling the update rule (4.5). In contrast, we use completely unspecific initialization with random values for all TF-bins in Case 3 and Case 4. The fifth column indicates, how we initialize the activations $\mathrm{H}^{(0)}$. In Case 1a, Case 1b, and Case 3 we use the score-based method described in Section 4.2.3, whereas we use completely unspecific initialization with a non-negative constant value in Case 2 and Case 4. Case 4 is our baseline, since the only side-information used is the number of components $R = C$.

4.3.3 Parameter Optimization

In Figure 4.4, we present the outcomes of two preliminary grid searches for a suitable working point of the parameters T and λ, both of which are influential for the NMFD convergence. On the one hand, we can assume the expressiveness of the NMFD model in (4.4) to increase with higher T. On the other hand, T should not greatly exceed the typical duration of the expected drum sound events. Otherwise, the chance to erroneously model rhythmically repeated events increases. The decay parameter λ is important in Case 1a, Case 1b and Case 3, where the initial NMFD activations $\mathrm{H}^{(0)}$ are populated with decaying impulses at the expected onset times in order to start activation updates near a global optimum.
Thus, we first sweep through different T to estimate an optimal value suited for drum sounds. In order to prevent overfitting, we conduct the grid search using the secondary dataset of isolated

drum sounds mentioned earlier (see Section 4.1.3). For each of the 693 single drum sound events, we first compute the STFT using blocksize N corresponding to a frame duration of approx. 46 ms, and hopsize H corresponding to approx. 11 ms. Then, we model the resulting magnitude spectrogram by running multiple NMFDs with $R = 1$ while varying $T \in [1 : 20]$ to monitor the average KL Divergence after $L^{\text{NMF}} = 30$ NMFD iterations. The resulting curves are averaged per drum instrument and displayed in Figure 4.4a, together with their global minima marked by circles. Notice that there are no great differences between distinct drum instruments and an optimal value is around $T \approx 7$.

In a second grid search, we additionally sweep through $0 \leq \lambda \leq 1$ and compare the idealized, decaying impulse with the the previously learned NMFD activations H by means of Pearson's correlation coefficient. The rationale is that we should try to initialize the activation slope of an informed NMFD as close as possible to the outcome of an unconstrained NMFD in order to improve convergence. In Figure 4.4b, the red circle marks the parameter combination yielding the maximum correlation, namely $T \approx 5$ and $\lambda \approx 0.55$. Note that high correlation values can be obtained with quite a variety of T and λ combinations, as indicated by the large, white area in the contour plot.

We repeated all source separation experiments with these different local optima of parameter combinations and found that the performance did not differ too drastically. Therefore, in the following experiments, we set $T := 5$ and $\lambda := 0.55$, leading to a maximum template duration of around 500 ms. The selectivity parameter of the α-Wiener filter is set to $\alpha := 1.0$.

Test case	**Estimator** $\tilde{\mathrm{V}}_c$	**Initial** $\mathrm{W}_t^{(0)}$	**Fixed** W_t	**Initial** $\mathrm{H}^{(0)}$
Case 0	Oracle	—	—	—
Case 1a	NMFD	Audio-based	yes	Score-based
Case 1b	NMFD	Audio-based	no	Score-based
Case 2	NMFD	Audio-based	no	Const. value
Case 3	NMFD	Random	no	Score-based
Case 4	NMFD	Random	no	Const. value

Table 4.1. Overview of test conditions in our baseline experiment.

4.3.4 Discussion of Baseline Results

In Figure 4.5, we present the results of our baseline experiment. The SDR, SIR and IPS values are obtained by averaging across all 60 test items for all three drum instruments. It is remarkable that the upper bound obtained in Case 0 already shows quite some variability and instrument dependency, which may be explained by the sub-optimality of the Wiener filtering approach as discussed in [134], as well as the usage of the mixture phase for the signal reconstruction as discussed in [38]. As expected, removing prior information leads to decreasing quality scores. The higher SIR range for kick drums is probably due to the more concentrated spectral distribution

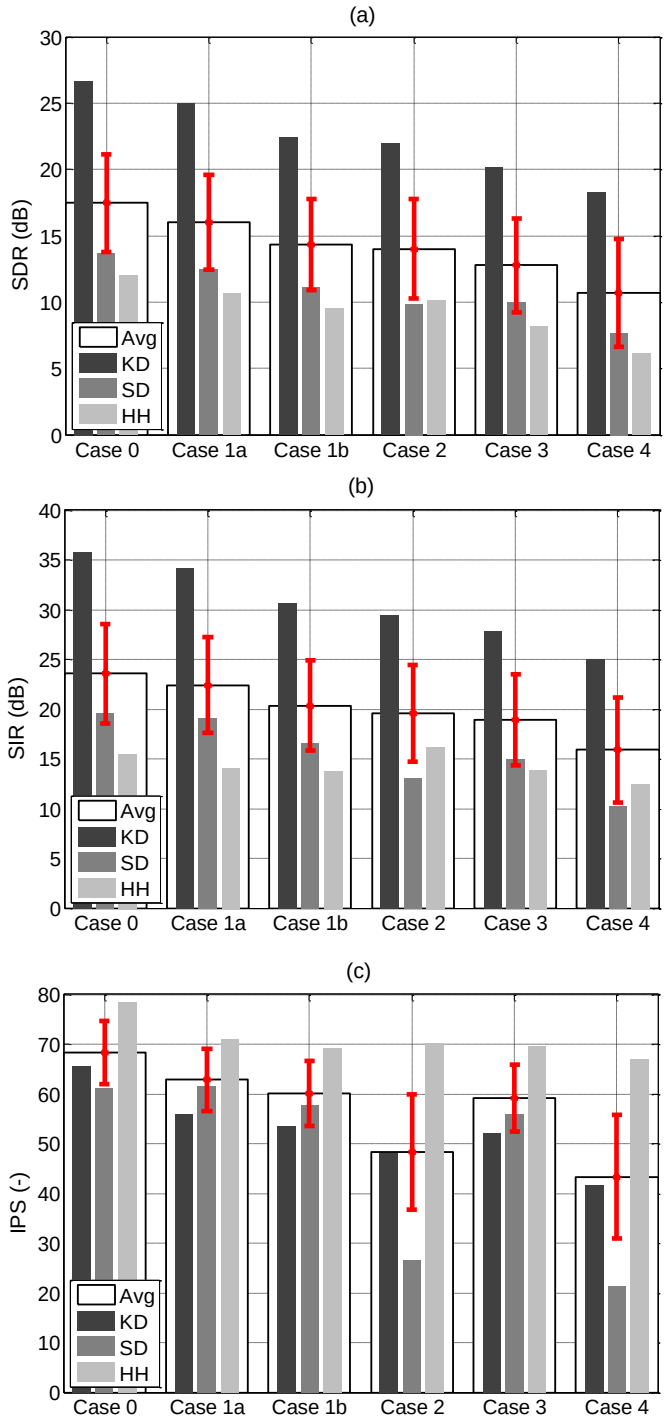

Figure 4.5. Baseline evaluation results showing separation quality for all instruments (Avg), kick drum (KD), snare drum (SD), and hi-hat (HH), attainable in the test cases from Table 4.1. **(a)** Average and standard deviation (in red) of the energy-related SDR metric. **(b)** Energy-based SIR metric. **(c)** Perceptually motivated IPS metric.

in the lower frequencies, whereas snare drum and hi-hat are more likely to overlap in the treble frequency region. Regarding the perceptually motivated IPS, it may be said that using score-informed constraints for NMFD (Case 3) is almost as good as the fully informed Case 1b. In conclusion, one should interpret these results carefully, especially since IPS is based on a learned mapping to user ratings that did not include drum sound items. Therefore, we recommend to listen to some of the component signals[21] in order to get a clearer impression of the attainable separation quality.

4.4 Separate and Restore

As we have seen from the systematic evaluation in Section 4.3, guiding the NMFD decomposition with score-based information (i.e., onsets for each drum instrument), or audio-based information (i.e., training examples of isolated drum sounds) yields similar separation quality. However, depending on the particular recording and instrument, we observe large variations in the quality scores. Closer inspection of test items with the lower scores revealed that the major problem is a substantial cross-talk from the hi-hat component into the kick drum and snare drum components. As described in Section 4.2.4, we apply soft masks (see (4.9) and (4.8)) in order to obtain component magnitude spectrograms V_c corresponding to individual drum instruments. We observed that the unwanted cross-talk artifacts are already apparent in the NMFD reconstructions $\tilde{\mathrm{V}}_c$ and thus cannot be be remedied by the subsequent α-Wiener filtering stage. Perceptually, the unwanted interference can roughly be divided in two subproblems. First, some items exhibited bleeding from hi-hat sounds with a long decay (e.g., open hi-hat) to segments of the snare drum or kick drum component that should not have any activity at all (i.e., silence between drum sound events). Second, we encounter spectral overlap during the transient attack of drum sound events, where concurrent drum sounds are a problem.
Interestingly, the somewhat unrealistic use of training examples for audio-informed initialization of $\mathrm{W}_t^{(0)}$ did not lead to substantial improvements. In Case 1b, we occasionally observed that properly initialized hi-hat magnitudes diffused into the kick drum and snare drum over the course of the NMFD iterations. This could only be countered by fixing the template spectrograms to the initial values in Case 1a. In the following sections, we introduce two restoration approaches that can be applied in isolation or combination to address the aforementioned problems.

4.4.1 Cascaded NMFD for Cross-Talk Attenuation

Despite potential corruption by cross-talk, the learned spectrogram templates are usually dominated by the target drum sound, while the cross-talk magnitude acts more like additive noise.

[21] Examples can be found on the accompanying website of this chapter, see `https://www.audiolabs-erlangen.de/resources/MIR/2016-IEEE-TASLP-DrumSeparation/`, last accessed June 14, 2018

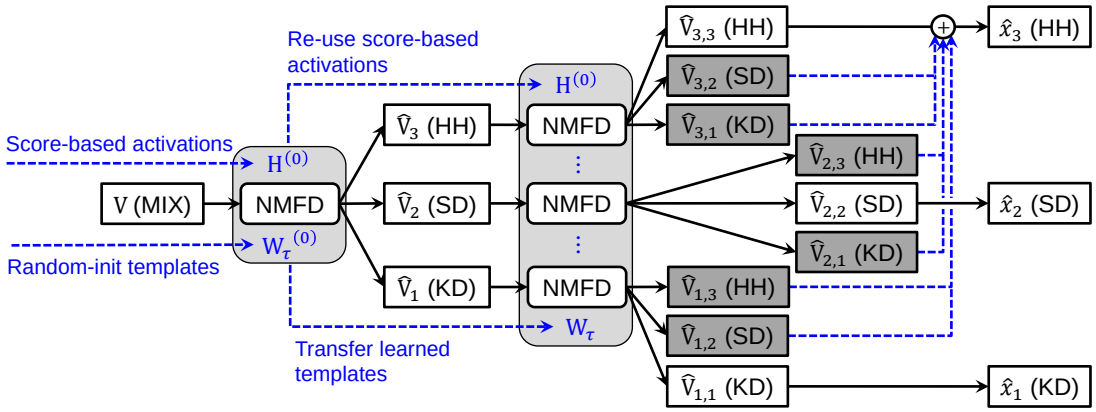

Figure 4.6. Cascaded NMFD decompositions that can be used to attenuate unwanted cross-talk between the initially extracted V_c (here shown for $c = 1, 2, 3$). The left half of the signal processing chain comprises the informed NMFD as well as the α-Wiener filtering as described in Section 4.2.2, 4.2.3, and 4.2.4. Indicated by the dashed blue arrows, the idea is to re-use the score-informed activations $\mathrm{H}^{(0)}$ and to transfer the templates W_t learned during the first NMFD process for initialization of the subsequent NMFD processes.

This observation leads us to a cascaded procedure that is based on applying secondary NMFD processes while retaining some of the side information. This time, instead of decomposing V, we apply NMFD to each of the V_c that were previously extracted via α-Wiener filtering. The idea is that the distinct spectral characteristics of the cross-talk artifacts will lead to a redistribution to the component that they belong to (usually the hi-hat). To enforce this behavior, we re-use the previously learned W_t as well as the score-informed initial $\mathrm{H}^{(0)}$ to initialize the templates and activations of the secondary NMFD processes as shown in Figure 4.6. After L^{NMF} iterations, we assign the resulting components (white rectangles) as new estimates for V_c. In our example, this means that $\hat{\mathrm{V}}_{1,1}$ is regarded as a kind of cross-talk reduced version of the kick drum component $\hat{\mathrm{V}}_1$. Similarly, $\hat{\mathrm{V}}_{2,2}$ is used to replace the snare drum component $\hat{\mathrm{V}}_2$. All other components are added to the hi-hat, as indicated by the dashed blue arrows leading to $\hat{x}_3$ in the upper right corner of Figure 4.6.

This simple procedure typically leads to a substantial attenuation of cross-talk, especially in segments where the target components should remain silent. In the next section, we describe a restoration procedure that can be used reduce the remaining cross-talk artifacts that might occur during attack phases.

4.4.2 Component Restoration using a Drum Sound Dictionary

In order to tackle the interference problem during attack sections, we propose a restoration procedure that is conceptually similar to the one introduced in [42]. The idea is to use a dictionary of cross-talk-free magnitude spectra obtained from isolated drum sounds. The rationale is that individual TF-bins in V_c might be corrupted and can potentially be restored if perfectly isolated

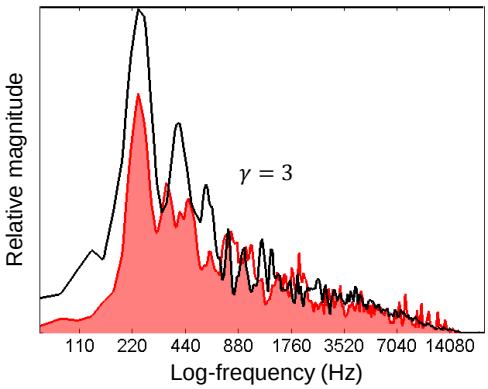

Figure 4.7. Example for query spectrum (red outline), matched dictionary spectrum (black outline) and element-wise minimum of both (light red). Both spectra are ℓ^1-normalized, the dictionary spectrum is multiplied with γ.

drum sound spectra are used as a stencil.

Similar to [68], we take $\hat{\mathrm{V}}_c$ extracted so far as a best estimate of the true target component. In our dictionary, we store ℓ^1-normalized spectra obtained by averaging across time in the magnitude spectrograms of drum sounds recorded in isolation. We apply a very basic matching procedure in order to retrieve spectra from the dictionary that are similar to the $\hat{\mathrm{V}}_c$. As a similarity measure, we employ Pearson's correlation coefficient between the component spectrum (obtained by averaging across all M frames of each $\hat{\mathrm{V}}_c$) and the dictionary entries. For each drum instrument, we select the best match from the dictionary and denote it as $\mathrm{V}^{\mathrm{Dict}} \in \mathbb{R}_{\geq 0}^{K \times 1}$. For the following considerations, we introduce an alternative representation $\mathrm{B}_c \in \mathbb{R}_{\geq 0}^{K \times M}$ of the targeted $\hat{\mathrm{V}}_c$ defined by

$$\mathrm{B}_c(k,m) := \frac{\hat{\mathrm{V}}_c(k,m)}{\mathrm{G}_c(m)}, \tag{4.12}$$

where $\mathrm{G}_c \in \mathbb{R}^{1 \times M}$ contains the element-wise ℓ^1-norm of each spectrogram frame of the previously extracted $\hat{\mathrm{V}}_c$. Furthermore, we introduce $\mathrm{B}^{\mathrm{Dict}} \in \mathbb{R}_{\geq 0}^{K \times M}$ as

$$\mathrm{B}^{\mathrm{Dict}} := \mathrm{V}^{\mathrm{Dict}} \cdot \mathrm{J}_{1,M}, \tag{4.13}$$

where $\mathrm{J}_{1,M} \in \mathbb{R}^{1 \times M}$ denotes an all-one matrix needed to replicate the best matching $\mathrm{V}^{\mathrm{Dict}}$ spectrum across all spectrogram frames. Consequently, we define a new $\hat{\mathrm{V}}_c$ by the following operation applied to each TF-bin

$$\hat{\mathrm{V}}_c(k,m) := \mathrm{G}_c(m) \cdot \min\left(\mathrm{B}_c(k,m), \gamma \cdot \mathrm{B}^{\mathrm{Dict}}(k,m)\right), \tag{4.14}$$

with the boost factor $\gamma \in \mathbb{R}_{\geq 0}$ applied to scale the dictionary-based spectrum. TF-bins whose magnitude exceeds the corresponding dictionary entry are trimmed by the minimum operation

Test case	Cascade	Dictionary	Comment
Cascade	Enabled	Disabled	Attenuates temporal cross-talk
Dictionary	Disabled	Enabled	Attenuates spectral cross-talk
Combined	Enabled	Enabled	Both methods in succession
Supervised	Conditional	Conditional	Best variant chosen by user

Table 4.2. Overview of test conditions in our restoration experiment.

on the right-hand side of (4.14). Afterwards, multiplication with the ℓ^1-norm reverts the previous normalization (see (4.12)). TF-bins with magnitudes smaller than the corresponding dictionary entries will not be affected by this process.
In Figure 4.7, we show in red an example spectrum in the attack frame of the snare component V_2 extracted in Case 3. We overlaid the best matching entry from the dictionary as a bold black curve. Now taking the TF-wise minimum of both ℓ^1-normalized spectra is indicated by the light red area. It can be seen that the γ-amplified dictionary spectrum is well above the averaged snare drum spectrum between 200 and 400 Hz, implying that the target magnitude will pass through this region. In the frequency regions above 7000 Hz, the substantial removal of cross-talk is visible.

4.4.3 Restoration Experiment

In this section, we present the experiments and results obtained for our proposed separate and restore strategies. To this end, we basically repeat the experiment from Section 4.3 but this time only consider Case 0 (oracle) and Case 3 (only score-based information) as reference values. Additionally, we define four new test conditions that are detailed in Table 4.2. In all additional cases, we start from the intermediate decomposition results V_c obtained in Case 3. Using the decomposition results of that condition allows us to assess the improvement that can be obtained by applying our two restoration procedures, either in isolation or in combination. In case of the dictionary-based restoration, we choose the boost parameter $\gamma = 3$ according to preliminary, informal quality comparisons. In the supervised case, we simulate human intervention by always selecting the method that yields the highest score per item and per drum instrument. In terms of real-world applicability, we think that it makes sense to leave this final decision up to the user. As outlined in Section 4.1, we typically expect a person to listen to results and pick the best variant, since our main application scenario is breakbeat separation for music production and remixing.
In Figure 4.8, we present the results using the same metrics as in Section 4.3.4. In terms of SDR, we only obtain improvements in the supervised case. With respect to the interference-related SIR, we observe that the scores are improving in all cases. For the perceptually motivated IPS, the scores in the supervised case almost reach the upper quality bound of Case 0. However, we advise the reader to take these figures with care and get a better impression from the audio

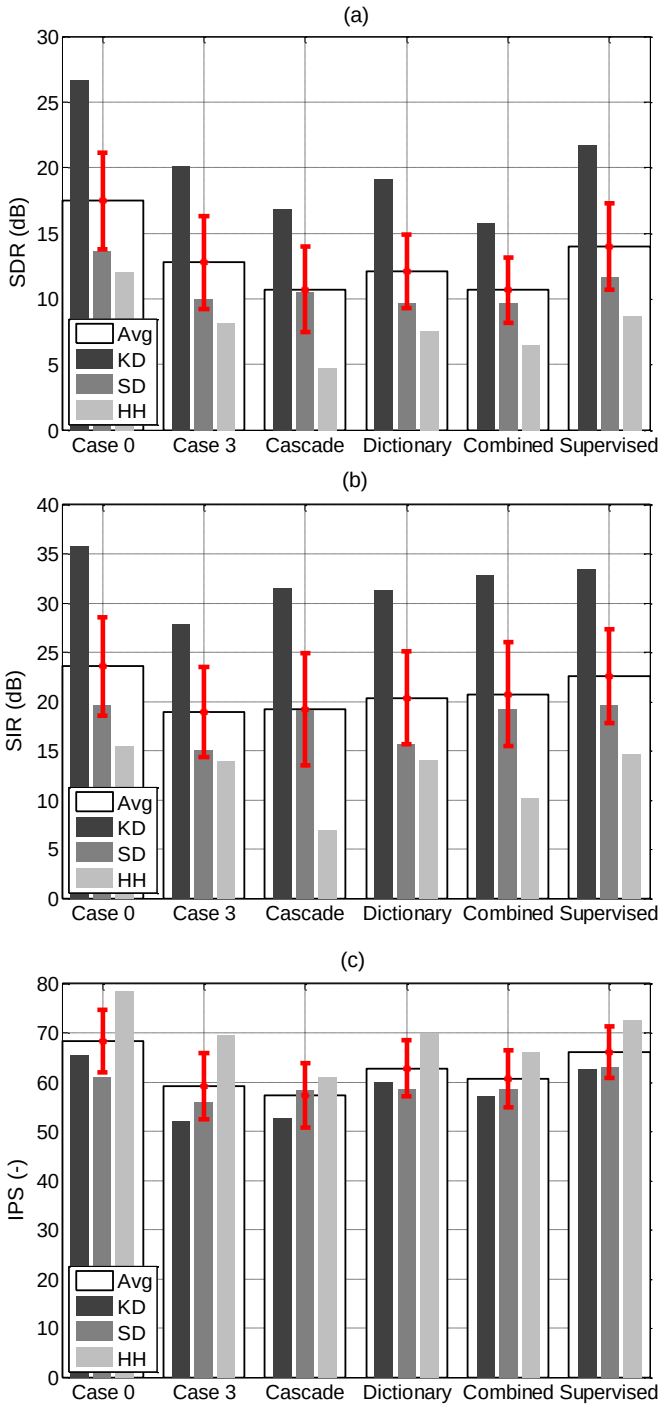

Figure 4.8. Evaluation results of our proposed restoration approaches. **(a)** Average and standard deviation (in red) of the energy-related SDR. **(b)** The energy-based SIR metric. **(c)** The perceptually motivated IPS metric.

examples on our accompanying website.

In Figure 4.9, we see the typical effect of applying the cascaded NMFD, the dictionary-restoration and both in succession. For the sake of visibility, the frequency axes have been resampled to

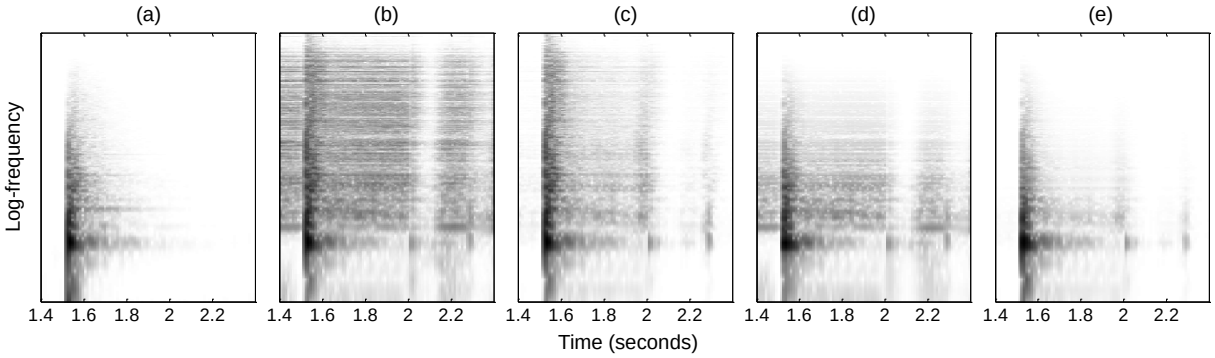

Figure 4.9. Example for the cross-talk attenuation with the proposed post-processing strategies. All plots show a detail zoom into the extracted magnitude spectrogram of component $c = 2$ (snare drum) in test item number 56. **(a)** The oracle component as given in the test corpus. **(b)** Separation result obtained by score-informed NMFD and α-Wiener filtering. **(c)** Result obtained by cascaded NMFD. **(d)** Result obtained by dictionary-based restoration. **(e)** Result obtained by successive application of cascaded NMFD and dictionary restoration.

logarithmic spacing and the magnitudes have are logarithmically compressed. In comparison to the oracle spectrogram excerpt in Figure 4.9a, the variant in Figure 4.9e (using both post-processing methods) optically yields the best cross-talk reduction. Again, we highly recommend to visit our accompanying website, where the shown test item is provided as an audio example.

4.5 Real-World Applicability

So far, we used a dataset of very clean drum recordings to create a controlled experimental setup. In this section, we illustrate that our proposed methods are also effective for separating real-world breakbeats, although some of our basic assumptions may be violated by this kind of drum solo recordings. Since the original recording conditions are generally unknown, we have no information about the acoustics of the studio and the equipment used at the time. Instead of linear superposition of the original sources, it is highly likely that we encounter convolutive, non-linear mixtures. However, our proposed methods are still able to achieve a reasonable separation quality, as can be heard in audio examples at the accompanying website[22].
In the second part of this section, we outline other application areas beyond breakbeat separation and discuss open problems.

[22]Please see `https://www.audiolabs-erlangen.de/resources/MIR/2016-IEEE-TASLP-DrumSeparation/`, last accessed June 14, 2018

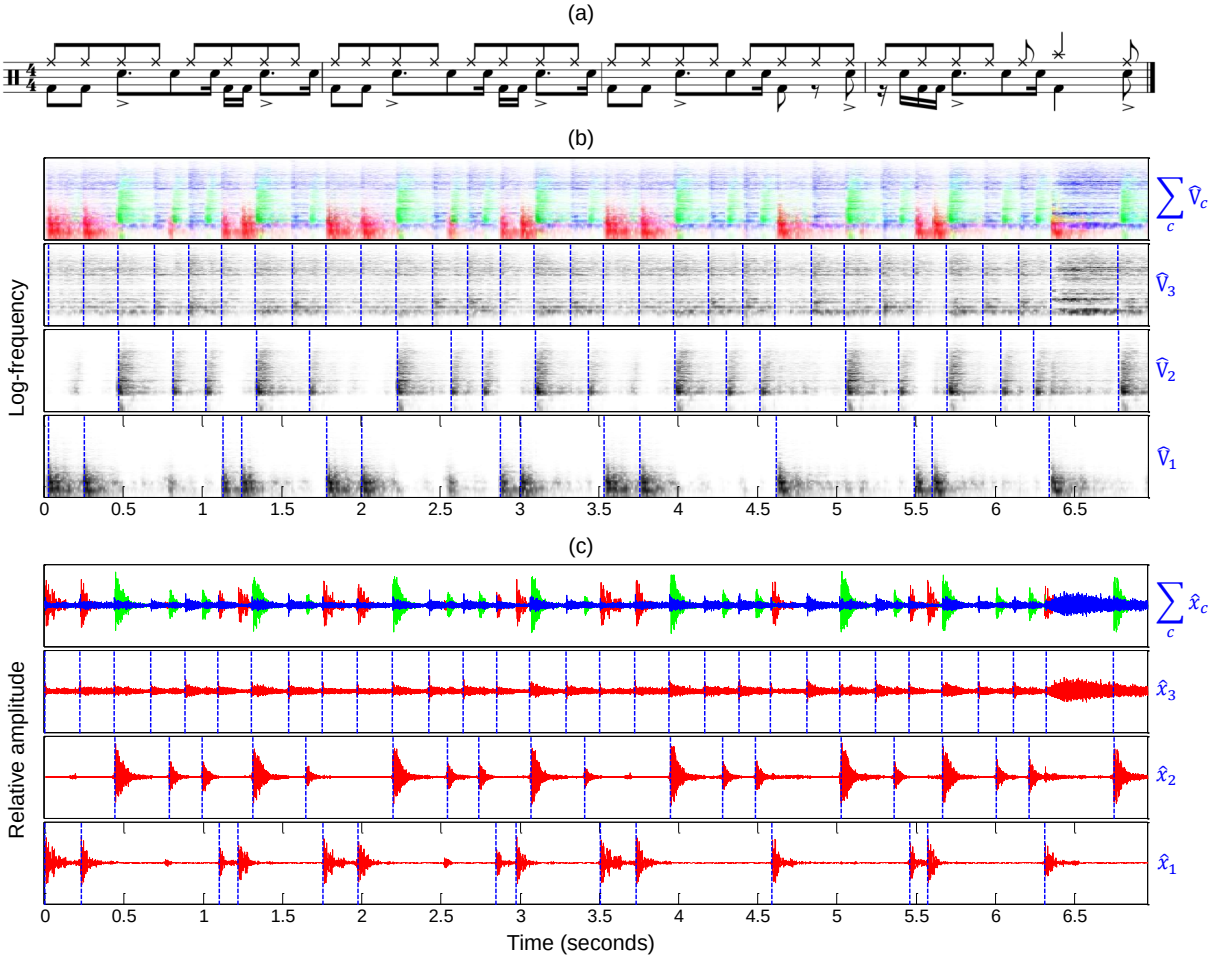

Figure 4.10. Reverse-engineered "Amen Break". **(a)** Time-aligned drum notation. **(b)** Color-coded mixture and component magnitude spectrograms $\hat{\mathrm{V}}_c$ on a logarithmic frequency and magnitude scale. **(c)** Color-coded mixture and component time-domain signals $\hat{x}_c$. Dashed blue lines correspond to the onset times of the respective drum instrument.

4.5.1 Reverse Engineering the Amen Break

In Figure 4.10, we show three different representations of the "Amen Break" (see Section 4.1). Figure 4.10a shows the drum notation of this breakbeat, based on the transcription in [163, pp. 325]. The drum solo mainly consists of kick drum, snare drum, and ride cymbal, playing a conventional funk drum pattern during the first two bars. In the second half of the third bar, things become more interesting: the listener's expectation of another repetition is deliberately broken and syncopation is used to shift accentuated snare hits into the offbeat.

For better readability, the single note events in the drum notation (Figure 4.10a) have been slightly repositioned to correspond to the onset events (dashed blue lines) in the spectrogram representation (Figure 4.10b), as well as the time-domain representation (Figure 4.10c). Both plots show the results of our decomposition method according to the signal model in Figure 4.2.

In addition, we use an alternative representation of the mixture (top panels in (b) and (c)), where we color-code the intensity values for kick drum in red, the snare drum in green and the ride cymbal in blue.
All components were extracted using the given transcription to impose score-based constraints (Case 3). Note that there is an accentuated note played on the crash cymbal shortly before the end of the fourth bar. To account for this additional instrument, the NMFD decomposition was performed with $C = 4$ components. For the sake of brevity, we mixed the ride and crash cymbal components back together afterwards. Since the kick drum component exhibited cross-talk from the ride cymbal, we applied the dictionary-based restoration described in Section 4.4.2 to this particular component.
From the $\hat{\mathrm{V}}_c$ in Figure 4.10b, we can see some interesting characteristics of the recording that would usually be hidden in the mixture. First, the snare component exhibits a sharp drop of energy below approx. 200 Hz. This indicates that the recording engineer used some sort of high pass filter for equalization on the snare microphone signal. In modern drum recording, one would usually also attenuate the lower frequency range of the ride cymbal signal in order to remedy buzzing and achieve a brilliant sound. Surprisingly, the ride cymbal in the "Amen Break" still exhibits energy in frequency regions below that snare drum cutoff frequency (almost in the kick drum range). This can clearly be seen in the color-coded mixture spectrogram. We can only speculate about the reasons for this characteristic, but it surely contributes to the special sound of this particular breakbeat.
In our accompanying website, we present three remixes of the "Amen Break" in order to demonstrate the creative capabilities that emerge once we have isolated all of its drum sound events. The application scenario can be described as cross-synthesis, meaning we can use single drum sounds extracted from the "Amen Break" to replace the drum hits of an unrelated target breakbeat.

4.5.2 Further Application Areas

Beyond breakbeat separation we want to discuss two further problems, where the proposed method could be beneficial.

Drum Processing: Post-processing of individual drum instruments plays an important role in professional music production. As already mentioned in Section 4.1.1, drum kits are usually recorded and processed in a multi-channel fashion, e.g., by equalizers, reverberation, dynamics processors, or drum replacement plug-ins[23]. However, proper microphone setup is not trivial and cross-talk (leakage) between channels is a major issue. In [122], an NMF-based approach for

[23] `http://www.drumagog.com/`, last accessed June 14, 2018

drum leakage suppression was proposed (which later went into the product Drumatom[24]). In [37], a real-time capable NMF variant suited for drum sound separation and leakage suppression was introduced. The methods proposed in this chapter are equally well suited for drum leakage suppression. Since we showed in Section 4.3.4 that audio-based side information can lead to good separation performance, training with isolated sounds of the expected drums would be beneficial. Having in mind that the single drum instruments are played in succession during sound-checks, it is quite realistic to fulfill that requirement in practice.

Music Restoration: More generally, the proposed method can be applied for music restoration. In that scenario, the aim is to extract sound events that correspond to the composition's individual notes, restore signal degradations like distortion and noise to create a high-fidelity, remastered version. Especially instruments that produces tones with discrete pitch, sharp attack, stable sustain, and a short release phase could potentially be modeled via NMFD. One concrete example could be score-informed restoration of historic piano or harpsichord recordings. A new mixture solely based on the extracted tones could have improved transients and attenuated noise. Of course, a robust automatic score alignment is required for high-quality results. Moreover, NMFD still is a simplification, since we neglect acoustic effects such as the room impulse response and sympathetic string resonances in the piano.

4.5.3 Open Issues

Throughout this chapter, we have assumed that a transcription of the target drum onsets is known. In practice, we might not even know the number C of drum instruments that are played in a drum recording. Again, asking the user to provide such information is one possible solution. Alternatively, there are many different approaches for automatic drum transcription (see [37] for an overview). Although most state-of-the-art methods are based on NMF, NMFD or PLCA, we think that transcription based on classification paradigms could provide a complementary perspective. There are promising methods for piano transcription that based on Boosting [207] or Recurrent Neural Networks [18] that could be potentially be adapted for drums.
Another issue is that the NMFD model is unsuited to account for additional, melodic sources. This problem might be approached by HPSS methods [52, 84, 155] or by modifying the NMFD objective in a TF-selective fashion [68, 211], effectively reducing the contribution of other sources than drum to the component updates.

[24] `http://drumatom.com/`, last accessed June 14, 2018

4.6 Conclusions and Further Notes

In this chapter, we presented a source separation scheme capable of decomposing breakbeat recordings into their constituent drum sound events. We conducted a baseline experiment to quantify the influential factors of the NMFD-based decomposition technique. We identified remaining problems of the core method and devised two restoration strategies that can be used to attenuate unwanted cross-talk resulting from imperfect decompositions. Since each of the proposed methods has benefits as well as limitations, depending on the application, we recommend involving the user in the pipeline by allowing a choice between alternative decompositions. Finally, we outlined application scenarios involving creative use of decomposed breakbeats. Future work will focus on deriving the required drum onset transcription semi-automatically. Furthermore, we want to extend to more challenging signal mixtures including those with bass or solo-instruments.

Chapter 5

The Separate and Restore Approach

The work in this chapter is mainly based on our contribution in [42].

Our goal is to improve the perceptual quality of signal components extracted in the context of music source separation. Specifically, we focus on decomposing polyphonic, mono-timbral piano recordings into the sound events that correspond to the individual notes of the underlying composition. Our separation technique is based on score-informed Non-Negative Matrix Factorization (NMF) that has been proposed in earlier works as an effective means to enforce a musically meaningful decomposition of piano music. However, the method still has certain shortcomings for complex mixtures where the tones strongly overlap in frequency and time. As the main contribution of this chapter, we propose a restoration stage based on refined Wiener filter masks to score-informed NMF. Our idea is to introduce notewise soft masks created from a dictionary of perfectly isolated piano tones, which are then adapted to match the timbre of the target components. A basic experiment with mixtures of piano tones shows improvements of our novel reconstruction method with regard to perceptually motivated separation quality metrics. A second experiment with more complex piano recordings shows that further investigations into the concept are necessary for real-world applicability.

5.1 Introduction

The goal of music source separation is to decompose a music recording into its constituent signal components [67, 187]. We focus on the special case of polyphonic, mono-timbral piano recordings, where we aim to extract sound events that correspond to the composition's individual notes. We assume that each sound event corresponding to a musical note can be characterized as a harmonic tone with constant pitch as well as a sharp attack, a stable sustain, and a release phase.

Moreover, the mixture signal is assumed to be a linear superposition of the isolated tones. This is, of course, a simplification since we neglect acoustic effects such as room impulse responses and sympathetic string resonances in the piano. With these assumptions, the decomposition of piano music into isolated tones constitutes a limited, yet challenging source separation task, which may pave the way to more complex scenarios.
One of the challenges is to improve the perceptual quality of the separated signals which may suffer from audible artifacts, depending on the complexity of the music, the recording conditions, as well as the decomposition technique. In this chapter, we follow the paradigm of score-informed Non-Negative Matrix Factorization (NMF) to *separate* the mixture magnitude spectrogram as described in [51, 67]. This procedure involves soft masks used to derive the targeted component magnitude spectrograms by Wiener filtering. The soft masks are usually obtained from multiplying suitable NMF templates and activations. In contrast to that, we propose to refine the soft masks on the basis of a timbre-adapted dictionary of isolated tones. We refer to this extension of the conventional method as *restore* approach and show that it can be beneficial for certain types of mixtures.
The remainder of this chapter is organized as follows: Section 5.2 provides a brief overview of related work, Section 5.3 reviews score-informed NMF, Section 5.4 describes our proposed restore approach, Section 5.5 discusses the experimental results and indicates directions for future work.

5.2 Related Work

Music source separation using score information has first been introduced in [219] and [132]. Related approaches used tensor factorization [33] or synthesized music for component initialization [83]. An important starting point for our work is the procedure for score-informed music decomposition described in [67], where the authors describe how to impose musically meaningful constraints on the components of a Non-Negative Matrix Factorization (NMF) without the need for a dedicated component training. The same principle is extended to note-wise decomposition in [51]. Other authors devised elaborate source-filter models to account for the time-varying spectral envelope of components with fixed pitch [56, 103]. In [37], a semi-adaptive NMF variant, which allows to efficiently capture the temporal evolution of component spectrograms, was proposed.
In [68], Ewert discussed the problems inherent to NMF decomposition in case of overlapping partials of the targeted components. He proposed to use the activations and gains computed by a first NMF stage to infer a time-frequency (TF) dependent weighting of the mixture magnitude spectrogram accounting for possible phase cancellations. He could show that this leads to more meaningful decompositions in a second NMF stage. The use of a weighted NMF had already been proposed by Virtanen in [211], where it was used as a means to fill gaps in the magnitude spectrogram that occurred due to binary masking of predominant pitched signal components. In [27], Cano et al. investigated the complex mutual influence of magnitude and phase on the quality

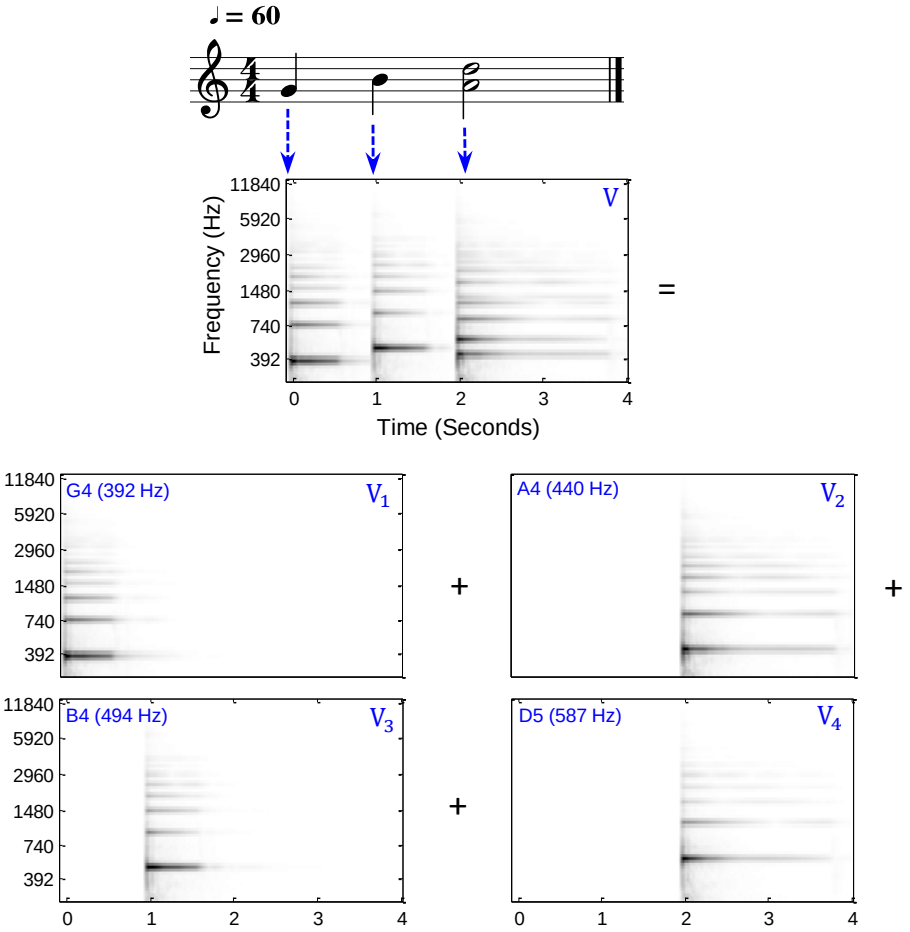

Figure 5.1. Artificial piano score used as illustrative example throughout the chapter. Our target is the extraction of the isolated magnitude spectrograms corresponding to the individual notes G4, A4, B4, and D5 (ordered by pitch instead of onset time).

of separated signals in source separation. Cano proposed to soften the additivity constraint in source separation and suggested to use instrument specific resynthesis approaches in [28].

5.3 Separate

In this section, we summarize the score-informed NMF approach as described in [51, 67]. In our signal model, we assume that the given piano recording x is a linear mixture of notewise audio events x_s, $s \in [1:S]$, where $S \in \mathbb{N}$ is the number of musical notes specified in the musical score (see our example score in Figure 5.1) such that $x := \sum_s x_s$. Let $\mathcal{X}(m,k)$, $m,k \in \mathbb{Z}$, be a complex-valued TF coefficient at the m^{th} time frame and k^{th} frequency bin of the Short-Time Fourier Transform (STFT) of our mixture signal x. Let $\mathrm{V} := |\mathcal{X}|^{\top} \in \mathbb{R}_{\geq 0}^{K \times M}$ be a transposed version of the mixture signal's magnitude spectrogram. Our objective is to decompose V into component

magnitude spectrograms V_s that correspond to the individual note events x_s. Ignoring possible phase issues, we assume that the additive relationship $\mathrm{V} := \sum_s \mathrm{V}_s$ is fulfilled (see Figure 5.1).

5.3.1 Music Decomposition via NMF

NMF can be used to decompose the magnitude spectrogram V into spectral basis functions (also called templates) encoded by the columns of $\mathrm{W} \in \mathbb{R}_{\geq 0}^{K \times R}$ and time-varying gains (also called activations) encoded by the rows of $\mathrm{H} \in \mathbb{R}_{\geq 0}^{R \times M}$ such that $\mathrm{V} \approx \mathrm{WH}$. NMF typically starts with a suitable initialization of matrices $\mathrm{W}^{(0)}$ and $\mathrm{H}^{(0)}$. Subsequently, these matrices are iteratively updated to adapt to V with regard to a suitable distance measure. In this work, we use the well-known update rules for minimizing the Kullback-Leibler Divergence [130] given by

$$\mathrm{W}^{(\ell+1)} = \mathrm{W}^{(\ell)} \odot \frac{\frac{\mathrm{V}}{\mathrm{W}^{(\ell)}\mathrm{H}^{(\ell)}}\mathrm{H}^{(\ell)^\top}}{\mathrm{J}\mathrm{H}^{(\ell)^\top}} \tag{5.1}$$

$$\mathrm{H}^{(\ell+1)} = \mathrm{H}^{(\ell)} \odot \frac{\mathrm{W}^{(\ell+1)^\top}\frac{\mathrm{V}}{\mathrm{W}^{(\ell+1)}\mathrm{H}^{(\ell)}}}{\mathrm{W}^{(\ell)^\top}\mathrm{J}} \tag{5.2}$$

for $\ell = 0, 1, 2, \ldots, L$ for some $L \in \mathbb{N}$. The symbol $\odot$ denotes element-wise multiplication and the division is also understood element-wise. Furthermore, $\mathrm{J} \in \mathbb{R}^{K \times M}$ denotes an all-one matrix.

5.3.2 Constraint Components via Score-Informed NMF

Proper initialization of $\mathrm{W}^{(0)}$ and $\mathrm{H}^{(0)}$ is an effective means to constrain the degrees of freedom in the NMF iterations and enforces convergence to a desired, musically meaningful solution. One possibility is to impose constraints derived from a time-aligned, symbolic representation (i.e., machine readable score) of the recording [33]. Three constraints can be obtained from the musical score. First, the rank R of the decomposition is chosen according to the number of unique musical pitches. Second, each column of $\mathrm{W}^{(0)}$ is initialized with a prototype harmonic overtone series reflecting the expected nature of a musical tone corresponding to the assigned musical pitch. Third, the rows of $\mathrm{H}^{(0)}$ are initialized as follows: A binary constraint matrix $\mathrm{C}_s \in \mathbb{R}^{R \times M}$ is constructed for each $s \in [1 : S]$, where C_s is 1 at entries that correspond to the pitch and temporal position of the s^{th} aligned note event and 0 otherwise. The union (OR-sum) of all C_s is then used as initialization of $\mathrm{H}^{(0)}$. With this initialization, each template obtained from iteration (5.1) typically corresponds to an average spectrum (usually ℓ^1-normalized [187]) of the corresponding musical pitch and each activation function obtained from (5.2) corresponds to the temporal amplitude envelope of all occurrences of that particular pitch throughout the recording.

5.4 Restore

Score-informed NMF as described in Section 5.3.2 yields a decomposition of V into musically meaningful templates $\mathrm{W}^{(L)}$ and activations $\mathrm{H}^{(L)}$. In the following, we discuss the issues inherent to the restoration of our targeted V_s from the components and introduce our extension to the conventional procedure.

5.4.1 Component Magnitude Spectrogram Reconstruction

As shown on the left hand side of Figure 5.2a, we can use the results of score-informed NMF to reconstruct magnitude spectrograms corresponding to individual note objects as $\mathrm{V}_s^{\mathrm{NMF}} \approx \mathrm{W}^{(L)} \left(\mathrm{H}^{(L)} \odot \mathrm{C}_s\right)$. In the conceptual illustration, the template and activation corresponding to the s^{th} tone are indicated by a hatched column and row, respectively. The binary activation in the corresponding C_s is visualized by a black box inside the hatched row. In order to obtain a time-domain signals from $\mathrm{V}_s^{\mathrm{NMF}}$, it is common practice to use the mixture phase information of the original STFT $\mathcal{X}$ and to invert the resulting modified STFTs via the signal reconstruction method from [96]. However, NMF-based models typically yield only a rough approximation of the original magnitude spectrogram, where spectral nuances may not be captured well. Therefore, the audio components reconstructed in this way may contain a number of audible artifacts. In order to better capture the temporal evolution of the spectral nuances, it is common practice to calculate soft masks that can be interpreted as a weighting matrix reflecting the contribution of the s^{th} tone to the original mixture V. The mask corresponding to the desired note event can be computed as $\mathrm{M}_c^{\mathrm{NMF}} := \mathrm{V}_s^{\mathrm{NMF}} \oslash \left(\mathrm{W}^{(L)}\mathrm{H}^{(L)} + \epsilon\right)$, where $\oslash$ denotes element-wise division and ϵ is a small positive constant to avoid division by zero. We obtain the masking-based estimate of the component magnitude spectrogram as $\mathrm{V}_s^{\mathrm{Mask}} := \mathrm{V} \odot \mathrm{M}_c^{\mathrm{NMF}}$. This procedure is also often referred to as Wiener filtering.

5.4.2 Difficult Mixtures

As discussed in [68], even the integration of score information might not suffice to separate certain mixtures. This is especially true in the case of mutually overlapping harmonics and transients. Our artificial example exhibits some of these problems. The decay of the two quarter notes (G4 and B4) is interfered by the attack transient of the subsequent notes. The last two notes (A4 and D5) are played simultaneously, so their attack transients overlap and some of their harmonics collide since they have very close center frequencies due to the harmonic relationship between the two notes (fourth interval). As can be seen in Figure 5.3b this leads to corrupted NMF templates. The note $s = 2$ (A4) exhibits a spurious peak around 600 Hz in a spectrogram frame that lies within the attack phase of $\mathrm{V}_s^{\mathrm{NMF}}$. The artifact is caused by "crosstalk" of the

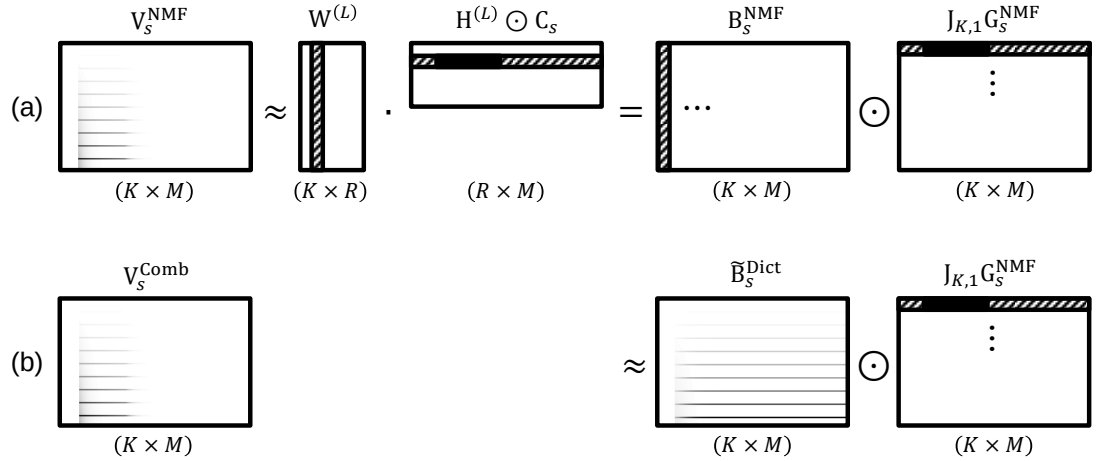

Figure 5.2. Conceptual illustration of **(a)** the procedures described in Section 5.4.1 and Section 5.4.2, and **(b)**, the restore approach described in Section 5.4.3.

first partial of D5 whose center frequency is 587 Hz and leads to deteriorated separation quality. For the following considerations, we introduce an alternative representation $\mathrm{B}_s \in \mathbb{R}_{\geq 0}^{K \times M}$ of the targeted V_s defined by

$$\mathrm{B}_s(k,m) := \frac{\mathrm{V}_s(k,m)}{\mathrm{G}_s(m)} \tag{5.3}$$

where $\mathrm{G}_s \in \mathbb{R}^{1 \times M}$ contains element-wise the ℓ^1-norm of each spectrogram frame of the original V_s. On the right hand side of Figure 5.2a, we show $\mathrm{B}_s^{\mathrm{NMF}}$ derived by application of (5.3) to $\mathrm{V}_s^{\mathrm{NMF}}$. Each spectrogram frame in $\mathrm{B}_s^{\mathrm{NMF}}$ is depicted as a replicate of the ℓ^1-normalized template corresponding to s. This seemingly redundant representation will become useful in Section 5.4.3, where we replace the potentially imperfect NMF templates by timbre-adapted and ℓ^1-normalized spectral templates taken from a dictionary.

5.4.3 Component Restoration Using a Note Dictionary

Inspired by the work of [33] and [83], we construct a dictionary of crosstalk-free magnitude spectrograms $\mathrm{V}_s^{\mathrm{Dict}}$ obtained from isolated piano tones. Via the score information, we select the appropriate tone and place it in the TF domain according to the onset position of the target note. However, the benefit of having an artifact-free $\mathrm{V}_s^{\mathrm{Dict}}$ comes at the expense that the dictionary tone is likely to differ in timbre from the target tone in the mixture. In order to adapt the timbral qualities of the dictionary without propagating potential errors in the NMF decomposition, we propose to transfer the spectral envelope of the target tone to the corresponding dictionary tone. Figure 5.3 illustrates this procedure for the note A4 ($s = 2$) in our artificial example. Similar to [68], we take $\mathrm{V}_s^{\mathrm{NMF}}$ as best estimate of the target component regardless of potential

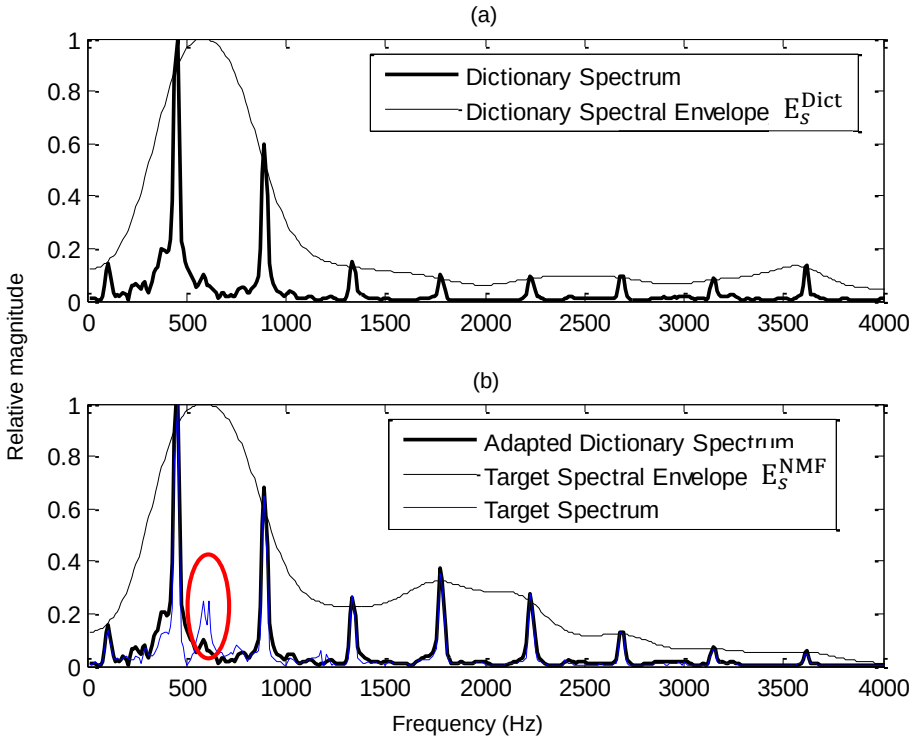

Figure 5.3. **(a)** Spectrogram frame (bold black curve) located in the attack phase of the dictionary tone representing the note A4. Its spectral envelope $\mathrm{E}_s^{\mathrm{Dict}}$ (thin black curve) is extracted via the true envelope method [173]. **(b)** Due to an imperfect NMF decomposition, a spurious peak is present around 600 Hz (marked by the red oval) in the target spectrum (blue curve). After an envelope transfer, the adapted dictionary spectrum follows nicely the target envelope $\mathrm{E}_s^{\mathrm{NMF}}$, but does not contain the artifact.

decomposition artifacts. Furthermore, we assume that a single estimate for the spectral envelope can be applied to the complete tone spectrogram. The estimate could e.g., be derived from an average spectrum. In our case, we expect the target component to dominate over potential cross-talk components in a spectral frame located in the attack phase of the tone. From that particular frame, we extract the spectral envelope $\mathrm{E}_s^{\mathrm{NMF}} \in \mathbb{R}^{K\times 1}$ using the so-called true envelope method as described by Röbel et al. in [173]. This procedure iteratively refines an estimate for the spectral envelope obtained by conventional cepstral liftering. Using the same method, we extract an estimate for the spectral envelope $\mathrm{E}_s^{\mathrm{Dict}}$ of the dictionary spectrogram. Then, we introduce the timbre-adapted dictionary note spectrogram as

$$\tilde{\mathrm{V}}_c^{\mathrm{Dict}} := \mathrm{V}_s^{\mathrm{Dict}} \odot \frac{\left(\mathrm{E}_s^{\mathrm{NMF}} \mathrm{J}_{1,M}\right)}{\left(\epsilon + \mathrm{E}_s^{\mathrm{Dict}} \mathrm{J}_{1,M}\right)} \tag{5.4}$$

where the division is understood element-wise and $\mathrm{J}_{1,M} \in \mathbb{R}^{1\times M}$ denotes an all-one matrix needed to replicate both spectral envelopes across all frames. Subsequently, we apply (5.3) to $\tilde{\mathrm{V}}_c^{\mathrm{Dict}}$ in order to obtain $\tilde{\mathrm{B}}_s^{\mathrm{Dict}}$ that we now use to replace the potentially corrupted $\mathrm{B}_s^{\mathrm{NMF}}$ as shown in Figure 5.2b. This way, we obtain novel estimates for the component spectrogram and

corresponding soft mask as

$$\mathrm{V}_s^{\mathrm{Comb}} := \tilde{\mathrm{B}}_s^{\mathrm{Dict}} \odot \left(\mathrm{J}_{K,1} \cdot \mathrm{G}_s^{\mathrm{NMF}}\right) \tag{5.5}$$

$$\mathrm{M}_c^{\mathrm{Comb}} := \mathrm{V}_s^{\mathrm{Comb}} \oslash \left(\epsilon + \sum_s \mathrm{V}_s^{\mathrm{Comb}}\right) \tag{5.6}$$

where $\mathrm{J}_{K,1} \in \mathbb{R}^{K\times 1}$ denotes an all-one matrix needed to replicate the corresponding note activation across all frequency bins. In short, the sequence of (5.4),(5.5), and (5.6) allows us to combine the NMF-based activations with ℓ^1-normalized dictionary spectra that have been adapted to match the timbre captured in the NMF-based templates. The resulting note component spectrogram is again obtained by Wiener filtering as $\mathrm{V}_s^{\mathrm{Prop}} := \mathrm{V} \odot \mathrm{M}_c^{\mathrm{Comb}}$.

5.5 Experiments

We conducted two source separation experiments using wellknown quality metrics to assess the possible improvements achievable with our proposed approach. First, we evaluated the separation of simplistic piano tone mixtures. Second, we tried to decompose more complex piano recordings into bass and treble component signals.

5.5.1 Dataset

Our first test set consisted of pair-wise piano tone combinations. We assigned MIDI pitch P_1 to the first tone in the mixture and defined it to be the interfering signal. Consequently, we assigned MIDI pitch P_2 to the second tone and defined it to be the target signal. Both P_1, P_2 were varied from MIDI pitch $P = 21$ (A0, 27.5 Hz) to $P = 108$ (C8, 4186 Hz) resulting in 7569 tone pairs (including unison intervals). The underlying single tone signals were recorded from a real piano, while the tone dictionary used for separation was synthesized using the Pianoteq[25] physical modeling plugin. Each tone pair was treated as individual test item, i.e. only $R = 2$ components were used.

The second test uses a subset of 11 MIDI files from the Saarland Music Data (SMD[26]) collection. SMD contains MIDI files for various classical piano pieces which were performed by students of the Hochschule für Musik Saar on a Yamaha Disklavier. The Disklavier stores all key and pedal movements performed by the pianist in an interpreted MIDI file that is suitable for synthesizing piano performances with expressive dynamics and timing. Following the experimental design in [68], we split the note events in each of the interpreted MIDI files into a bass (MIDI pitch $P < 60$) and a treble set (MIDI pitch $P \geq 60$). We again used Pianoteq to synthesize the note sequences

[25] `https://www.pianoteq.com/`, last accessed June 14, 2018

[26] `https://www.audiolabs-erlangen.de/resources/MIR/SMD/midi`, last accessed June 14, 2018

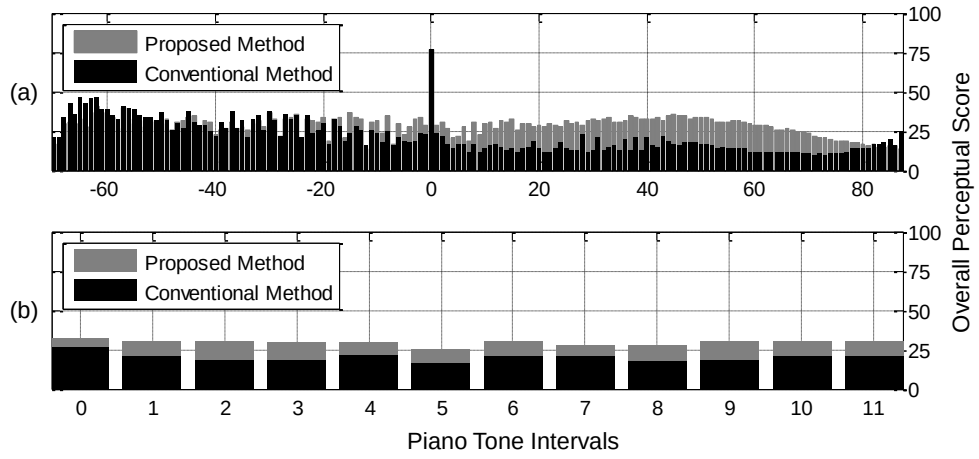

Figure 5.4. The Overall Perceptual Score (OPS) [62, 208] computed for separated piano tone mixtures with the conventional method (black bars) vs. the proposed method (gray bars). **(a)** OPS averaged over interval classes. **(b)** OPS averaged over absolute, octave-wrapped interval classes.

for the bass and treble set individually. The bass tones were defined to be the interferer and the treble tones the target, respectively. The superposition of both yielded our mixture signal.
All test files had 44.1 kHz sampling rate, the STFT was computed with a blocksize of approx. 46.4 ms and a hopsize of approx. 5.8 ms. The number of NMF iterations was set to $L = 30$. For each test item, we used $\mathrm{V}_s^{\mathrm{Mask}}$ as magnitude spectrogram representing the conventional approach and $\mathrm{V}_s^{\mathrm{Prop}}$ as magnitude spectrogram representing proposed reconstruction. We employed the PEASS Toolkit [62, 208] in order to evaluate the quality of the separated audio signals obtained from application of the conventional and the proposed method. From the available metrics, we focused on the perceptually-motivated Overall Perceptual Score (OPS) and used the objective Source-Distortion Ratio (SDR) to complement the evaluation.

5.5.2 Results and Discussion

From the first experiment, we derived two interval related evaluations. First, we aggregated the quality measures to the interval $\Delta P_{1,2} := P_2 - P_1$ between the two piano tones in each mixture and averaged the respective measurements. Second, we applied the same result aggregation, this time mapping to 12 absolute, octave-agnostic interval classes $\tilde{\Delta} P_{1,2} := (\Delta P_{1,2} \mod 12)$. Figure 5.4a shows that the restore approach surpasses the conventional Wiener filtering approach in terms of OPS mostly for positive intervals, i.e., where the MIDI pitch of the target is above the MIDI pitch of the interferer. Interestingly, this trend can not be observed for the SDR. Instead, the improvements are more evenly distributed and only decrease for very wide intervals regardless if they are positive or negative. Figure 5.4b shows that the proposed restore approach is approx. 9.5 OPS points ahead of the conventional approach if we ignore octave information. The average SDR improvement in that case amounts to approx. 0.7 dB.

We obtained very mixed results in our second experiment with realistic piano performances. On average, we achieved an OPS of 38.06 (SDR 10.16 dB) using conventional Wiener filtering, while our proposed restore approach yielded a tiny OPS increase to 38.32 (SDR 10.25 dB). Unfortunately, there is no consistent improvement across the test items, roughly half of them exhibit lower quality metrics compared to the conventional approach. From inspection of selected examples, we believe that this might be related to inferior separation of tones with a long sustain phase. Since we transfer the spectral envelope taken from the tone's attack, the sustain phase might diverge from the desired target over time. Audio examples covering positive and negative separation results are available online[27].

5.6 Conclusions and Further Notes

We presented a method for post-processing score-informed music decomposition by means of refined soft masks based on a dictionary of timbre-adapted piano tone spectrograms. In simple tone mixtures, this step attenuates the mutual interference between components. Despite mixed experimental results, we see potential in further developing the principal concept of our separate and restore approach. One obvious possibility would be an interval-selective application of the proposed method. Another possible direction is to investigate dedicated processing of percussive and harmonic NMF components to remedy some of the remaining problems related to unsatisfactory separation of complex mixtures. In the future, we want to further enhance this concept and investigate its applicability to other source separation tasks, such as drum sound separation from drum recordings.

[27] `http://www.audiolabs-erlangen.de/resources/MIR/2015-WASPAA-SeparateAndRestore/`, last accessed June 14, 2018

Chapter 6

Transient Restoration in Signal Reconstruction

The work in this chapter is mainly based on our contribution in [38].

Our goal is to improve the perceptual quality of transient signal components extracted in the context of music source separation. Many state-of-the-art techniques are based on applying a suitable decomposition to the magnitude of the Short-Time Fourier Transform (STFT) of the mixture signal. The phase information required for the reconstruction of individual component signals is usually taken from the mixture, resulting in a complex-valued, modified STFT (MSTFT). There are different methods for reconstructing a time-domain signal whose STFT approximates the target MSTFT. Due to phase inconsistencies, these reconstructed signals are likely to contain artifacts such as pre-echos preceding transient components. In this chapter, we propose a simple, yet effective extension of the iterative signal reconstruction procedure by Griffin and Lim to remedy this problem. In a first experiment, under laboratory conditions, we show that our method considerably attenuates pre-echos while still showing similar convergence properties as the original approach. A second, more realistic experiment involving score-informed audio decomposition shows that the proposed method still yields improvements, although to a lesser extent, under non-idealized conditions.

6.1 Introduction

Music source separation aims at decomposing a polyphonic, multi-timbral music recording into component signals such as singing voice, instrumental melodies, percussive instruments, or individual note events occurring in a mixture signal [67]. Besides being an important step in many

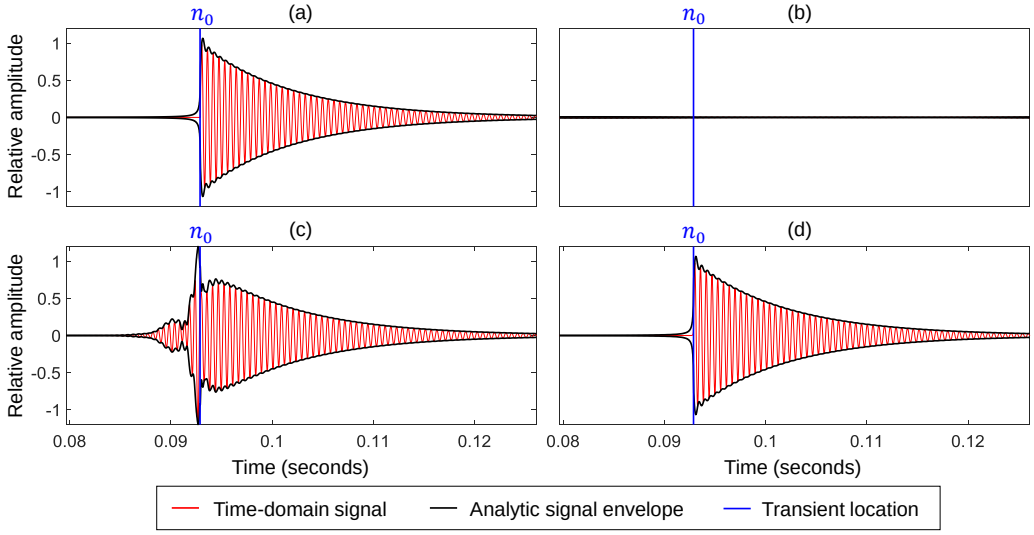

Figure 6.1. Illustration of the transient restoration. **(a)** Target component signal, an exponentially decaying sinusoid preceded by silence. **(b)** Reconstruction using zero phase. Due to destructive interference, the overall amplitude seemingly decreased to silence. **(c)** Reconstruction after 200 GL iterations, exhibiting pronounced transient smearing. **(d)** Reconstruction after 200 iterations of the proposed transient restoration method. The bottom legend applies to all plots, n_0 denotes the transient position.

music analysis and retrieval tasks, music source separation is also a fundamental prerequisite for applications such as music restoration, upmixing, and remixing. For these purposes, high fidelity in terms of perceptual quality of the separated components is desirable. The majority of existing separation techniques work on a time-frequency (TF) representation of the mixture signal, often the Short-Time Fourier Transform (STFT). The target component signals are usually reconstructed using a suitable inverse transform, which in turn can introduce audible artifacts such as musical noise, smeared transients or pre-echos, as exemplified in Figure 6.1c.

In order to better preserve transient signal components, we propose in this chapter a simple, yet effective extension to the signal reconstruction procedure by Griffin and Lim [96]. The original method iteratively estimates the phase information necessary for time-domain reconstruction from a magnitude STFT (STFTM) by going back and forth between the STFT and the time-domain, only updating the phase information, while keeping the STFTM fixed. Our proposed extension manipulates the intermediate time-domain reconstructions in order to attenuate the pre-echos that potentially precede the transients.

We conduct two kinds of evaluations in an audio decomposition scenario, where our objective is to extract isolated drum sounds from polyphonic drum recordings. To this end, we use a publicly available test set that is enriched with all necessary side information, such as the true "oracle" component signals and their precise transient positions. In the first experiment, under laboratory conditions, we make use of all side-information in order to focus on evaluating the benefit of our proposed method for transient preservation in signal reconstruction. Under these idealized conditions, we can show that our proposed method considerably attenuates pre-echos while still

exhibiting similar convergence properties as the original method. In the second experiment, we employ a state-of-the-art decomposition technique [133, 185] with score-informed constraints [67] to estimate the component signal's STFTM from the mixture. Under these more realistic conditions, our proposed method still yields improvements yet to a lesser extent than in the idealized scenario.
The remainder of this chapter is organized as follows: Section 6.2 provides a brief overview of related work before Section 6.3 introduces our new method. Section 6.4 details and discusses the experimental evaluation under laboratory conditions. Section 6.5 describes a more realistic application and evaluation of our proposed method in conjunction with score-informed audio decomposition. Finally, in Section 6.6 we conclude and indicate directions for future work.

6.2 Related Work

Three research fields are important for our work: First, a number of publications on signal reconstruction and transient preservation are related and relevant for our proposed restoration method. Second, papers on score-informed audio decomposition (i.e., source separation) provide the basis for deploying our method in a real-world application.

6.2.1 Signal Reconstruction

The problem of signal reconstruction, also known as magnitude spectrogram inversion or phase estimation is a well researched topic. In their classic paper [96], Griffin and Lim proposed the so-called LSEE-MSTFTM algorithm (denoted as GL throughout this chapter) for iterative, blind signal reconstruction from modified STFT magnitude (MSTFTM) spectrograms. In [126], Le Roux et al. developed a different view on this method by describing it using a TF consistency criterion. By keeping the necessary operations entirely in the TF domain, several simplifications and approximations could be introduced that lower the computational load compared to the original procedure. Since the phase estimates obtained using GL can only converge to local optima, several publications were concerned with finding a good initial estimate for the phase information [128, 229]. Sturmel and Daudet [196] provided an in-depth review of signal reconstruction methods and pointed out unsolved problems. An extension of GL with respect to convergence speed was proposed in [162]. Other authors tried to formulate the phase estimation problem as a convex optimization scheme and arrived at promising results hampered by high computational complexity [197]. Another work [148] was concerned with applying the spectrogram consistency framework to signal reconstruction from wavelet-based magnitude spectrograms.

6.2.2 Transient Preservation

The problem of transient preservation has been extensively addressed in the field of perceptual audio coding, where pre-echo artifacts can occur ahead of transient signal components. Pre-echos are caused by the use of relatively long analysis and synthesis windows in conjunction with coding-related modification of TF-bins such as quantization of spectral magnitudes according to a psycho-acoustic model. It can be considered as state-of-the-art to use block-switching to account for transient events [59]. An interesting approach was proposed in [104], where spectral coefficients are encoded by linear prediction along the frequency axis, automatically reducing pre-echos. Other authors proposed to decompose the signal into transient and residual components and use optimized coding parameters for each stream [151]. In [93], the authors proposed a scheme that unifies iterative signal reconstruction (see Section 6.2.1) and block-switching in the context of audio coding. Transient preservation has also been investigated in the context of time-scale modification methods based on the phase-vocoder [53]. In addition to an optimized treatment of the transient components, several authors follow the principle of phase-locking or re-initialization of phase in transient frames [57, 172].

6.2.3 Score-informed Audio Decomposition

The majority of music source separation techniques operate on a TF representation of the mixture signal. It is common practice to compute the mixture signal's STFT and apply suitable decomposition techniques (e.g., Non-Negative Matrix Factorization (NMF)) to the corresponding magnitude spectrogram. This yields an MSTFTM, ideally representing the isolated target signal component. The corresponding time-domain signal is usually derived by using the original phase information and applying signal reconstruction methods.
When striving for good perceptual quality of the separated target signals, many authors propose to impose score-informed constraints on the decomposition [33, 66, 67]. This has the advantage that the separation can be guided and constrained by information on the approximate location of component signals in time (onset, offset) and frequency (pitch, timbre). A few studies deal with source separation of strongly transient signals such as drums [7, 37]. Usage of the Non-Negative Matrix Factor Deconvolution (NMFD) for drum sound separation was first proposed in [185]. Later works applied it to drum sound detection using sparseness constraints [133] as well as regularisation in [174]. Others authors focus on the separation of harmonic vs. percussive components [29, 52, 71]. The importance of phase information for source separation quality is discussed in [27].

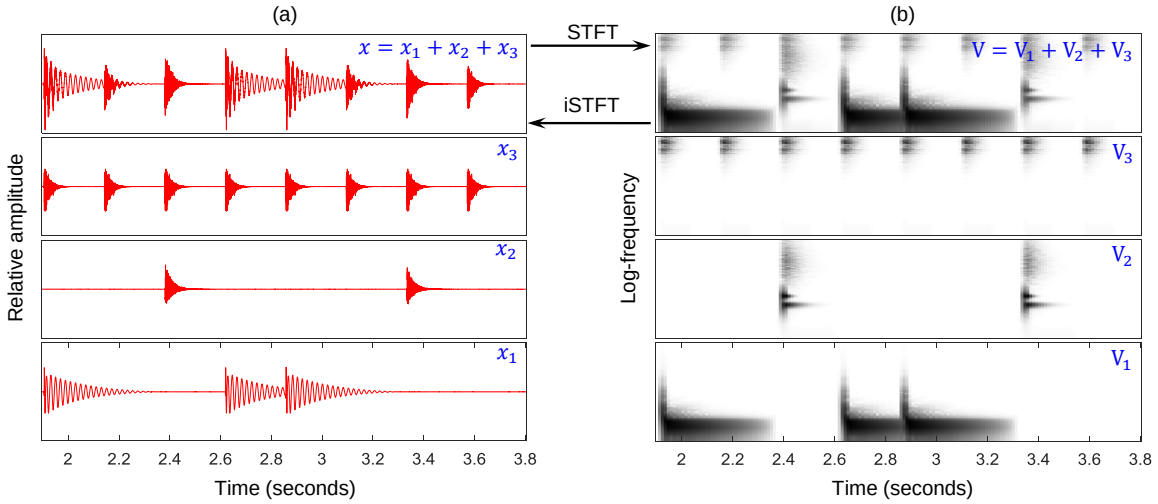

Figure 6.2. Illustration of our signal model. **(a)** Mixture signal x is the sum of $C = 3$ component signals x_c, each containing sequences of synthetic drum sounds sampled from a Roland TR 808 drum machine (x_1: kick drum, x_2: snare drum, x_3: hi-hat). **(b)** TF representation of the mixture's magnitude spectrogram V and $C = 3$ component magnitude spectrograms V_c. For better visibility, the frequency axis and the magnitudes are on a logarithmic scale.

6.3 Transient Restoration

In the following, we first fix our notation and signal model and describe the employed signal reconstruction method. Afterward, we introduce our novel extension for transient preservation in the GL method and provide an illustrative example.

6.3.1 Notation and Signal Model

We consider the real-valued, discrete time-domain signal $x : \mathbb{Z} \to \mathbb{R}$ to be a linear mixture $x := \sum_{c=1}^{C} x_c$ of $C \in \mathbb{N}$ component signals x_c corresponding to individual instruments. As shown in Figure 6.2a, each component signal contains at least one transient audio event produced by the corresponding instrument (in our case, by striking a drum). Furthermore, we assume that we have a symbolic transcription available that specifies the onset time (i.e., transient position) and instrument type for each of the audio events. From that transcription, we derive the total number of onset events S as well as the number of unique instruments C. Our aim is to extract individual component signals x_c from the mixture x as shown in Figure 6.2. For evaluation purposes (see Section 6.4), we assume to have the oracle component signals x_c available.
We decompose x in the TF-domain, to this end we employ STFT as follows. Let $\mathcal{X}(m, k)$ be a complex-valued TF coefficient at the m^{th} time frame and k^{th} spectral bin. The coefficient is computed by

$$\mathcal{X}(m,k) := \sum_{n=0}^{N-1} x(n + mH)w(n)\exp(-2\pi ikn/N), \tag{6.1}$$

where $w : [0 : N-1] \to \mathbb{R}$ is a suitable window function of blocksize $N \in \mathbb{N}$, and $H \in \mathbb{N}$ is the hop size parameter. The number of frequency bins is $K = N/2$ and the number of spectral frames M is determined by the available signal samples. For simplicity, we also write $\mathcal{X} = \mathrm{STFT}(x)$. Following [126], we call $\mathcal{X}$ a consistent STFT since it is a set of complex numbers which has been obtained from the real time-domain signal x via (6.1). In contrast, an inconsistent STFT is a set of complex numbers that was not obtained from a real time-domain signal. From $\mathcal{X}$, the magnitude spectrogram $\mathcal{A}$ and the phase spectrogram φ are derived as

$$\mathcal{A}(m,k) := |\mathcal{X}(m,k)|\,, \tag{6.2}$$

$$\varphi(m,k) := \angle\mathcal{X}(m,k), \tag{6.3}$$

with $\varphi(m,k) \in [0, 2\pi)$.
Let $\mathrm{V} := \mathcal{A}^\top \in \mathbb{R}_{\geq 0}^{K\times M}$ be a non-negative matrix holding a transposed version of the mixture's magnitude spectrogram $\mathcal{A}$. Our objective is to decompose V into component magnitude spectrograms V_c that correspond to the distinct instruments as shown in Figure 6.2b. For the moment, we assume that some oracle estimator extracts the desired $\mathcal{A}_c := \mathrm{V}_c^\top$. One possible approach to estimate the component magnitudes using a state-of-the-art decomposition technique will be described in Section 6.5. In order to reconstruct a specific component signal x_c, we set $\mathcal{X}_c := \mathcal{A}_c \odot \exp(i\varphi_c)$, where $\mathcal{A}_c = \mathrm{V}_c^\top$ and φ_c is an estimate of the component phase spectrogram. It is common practice to use the mixture phase information φ as an estimate for φ_c and to invert the resulting MSTFT via the LSEE-MSTFT reconstruction method from [96]. The method first applies the inverse Discrete Fourier Transform (DFT) to each spectral frame in $\mathcal{X}_c$, yielding a set of intermediate time signals y_m, with $m \in [0 : M-1]$, defined by

$$y_m(n) := \frac{1}{N}\sum_{k=0}^{N-1} \mathcal{X}_c(m,k)\exp(2\pi i k n/N), \tag{6.4}$$

for $n \in [0 : N-1]$ and $y_m(n) := 0$ for $n \in \mathbb{Z} \setminus [0 : N-1]$. Second, the least squares error reconstruction is achieved by

$$x_c(n) := \frac{\sum_{m\in\mathbb{Z}} y_m(n - mH)w(n - mH)}{\sum_{m\in\mathbb{Z}} w(n - mH)^2}, \tag{6.5}$$

$n \in \mathbb{Z}$, where the analysis window w is re-used as synthesis window. Please note that LSEE-MSTFT should not be confused with LSEE-MSTFTM (called GL in this work) that extends the signal reconstruction with iterative phase estimation (cf. Algorithm 1). In the following, for the sake of brevity, we will use $x_c = \mathrm{iSTFT}(\mathcal{X}_c)$ as short form for the application of (6.4) and (6.5).

6.3.2 Proposed Algorithm

Since we construct the MSTFT $\mathcal{X}_c$ in the TF domain, we have to consider that it may be an inconsistent STFT, i.e., there may not exist a real time-domain signal x_c fulfilling $\mathcal{X}_c = \mathrm{STFT}(x_c)$. Intuitively speaking, the complex relationship between magnitude and phase is likely corrupted as soon as the magnitude in certain TF-bins is modified. In practice, this inconsistency can lead to transient smearing and pre-echos in x_c, especially for large N.
To remedy this problem, we propose to iteratively minimize the inconsistency of $\mathcal{X}_c$ by the following extension (denoted as TR) of the GL procedure [96]. For the moment, let's assume that $\mathcal{X}_c$ contains precisely one transient onset event, whose exact location in time n_0 is known. Now, we introduce the iteration index $\ell = 0, 1, 2, \ldots L \in \mathbb{N}$. Given $\mathcal{A}_c$ and some initial phase estimate $\varphi_c^{(0)}$, we introduce the initial STFT estimate of the target component signal $\mathcal{X}_c^{(0)} := \mathcal{A}_c \odot \exp(i\varphi_c^{(0)})$ and apply Algorithm 1.

Algorithm 1: Transient Restoration (TR).

Input: $\varphi_c^{(0)}$ and $\mathcal{X}_c^{(0)} := \mathcal{A}_c \odot \exp\left(i\varphi_c^{(0)}\right)$

for $\ell = 0, 1, 2, \ldots, L-1$ **do**

- $x_c^{(\ell+1)} := \mathrm{iSTFT}\left(\mathcal{X}_c^{(\ell)}\right)$ via (6.4) and (6.5)
- Enforce $x_c^{(\ell+1)}(n) := 0$ for $n \in \mathbb{Z}, n < n_0$

} Intermediate reconstruction

- $\varphi_c^{(\ell+1)} := \angle\mathrm{STFT}\left(x_c^{(\ell+1)}\right)$ via (6.1) and (6.3)
- $\mathcal{X}_c^{(\ell+1)} := \mathcal{A}_c \odot \exp\left(i\varphi_c^{(\ell+1)}\right)$

} Phase update

end

Output: $x_c := x_c^{(L)}$.

The crucial point of our proposed extension is the second step in the intermediate reconstruction which enforces transient constraints in the GL procedure. Figure 6.1 illustrates our proposed method with the target component signal in red, overlaid with the envelope of its analytic signal in Figure 6.1a. The example signal exhibits transient behavior around n_0 (blue line) when the waveform transitions from silence to an exponentially decaying sinusoid. Figure 6.1b shows the time-domain reconstruction obtained from the iSTFT with $\varphi_c^{(0)} = 0$ (i.e., zero phase for all TF-bins). Through destructive interference of overlapping frames, the transient is completely destroyed, the amplitude of the sinusoid is strongly decreased and the envelope looks nearly flat. Figure 6.1c shows the reconstruction with pronounced transient smearing after $L = 200$ GL iterations. Figure 6.1d shows that the restored transient after $L = 200$ iterations of the proposed method is much closer to the original signal. In real-world recordings, there usually exist multiple

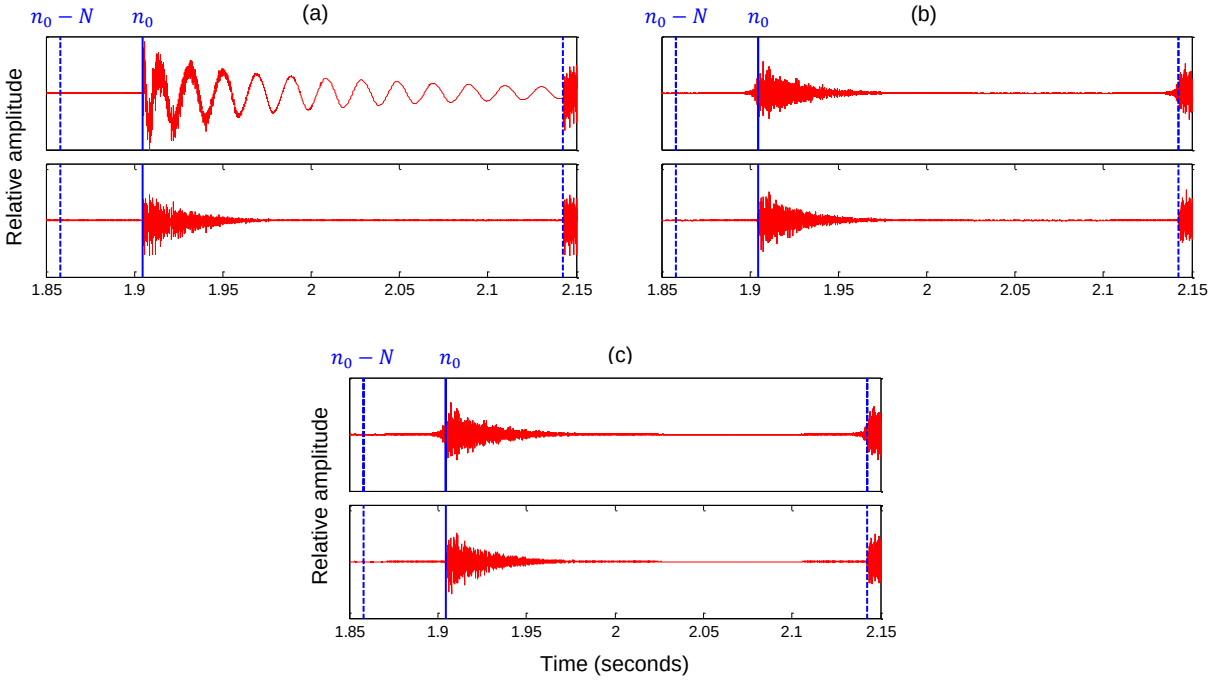

Figure 6.3. Different hi-hat component signals of our example drum loop. The transient position n_0 is given by the solid blue line, the excerpt boundaries by the dashed blue lines. **(a)** Mixture signal (top) vs. oracle hi-hat signal (bottom). **(b)** Hi-hat signal in Case 2, reconstruction after $L = 200$ iterations of GL (top) vs. TR (bottom). **(c)** Hi-hat signal in Case 4, reconstruction after $L = 200$ iterations of GL (top) vs. TR (bottom). Since the NMFD decomposition works very well for our example drum loop, there is almost no noticeable visual difference between (b) and (c).

transient onsets event throughout the signal. In this case, one may apply the proposed method to signal excerpts localized between consecutive transients (resp. onsets) as shown in Figure 6.3.

6.4 Evaluation under Laboratory Conditions

For evaluation, we compared the conventional GL reconstruction with our proposed TR method under two different initialization strategies for $\mathcal{X}_c^{(0)}$. In the following, we describe the used dataset, the test item generation, and our evaluation metrics.

6.4.1 Dataset

In principle, we follow the evaluation approach from [27]. In all our experiments, we use the publicly available "IDMT-SMT-Drums" dataset[28]. In the "WaveDrum02" subset, there are 60 drum loops, each given as perfectly isolated single track recordings (i.e., oracle component signals) of the three instruments kick drum, snare drum, and hi-hat. All 3×60 recordings are in

[28] `http://www.idmt.fraunhofer.de/en/business_units/smt/drums.html`, last accessed June 14, 2018

Test case	Initial phase estimate	Fixed magnitude estimate
Case 1	$\varphi_c^{(0)} := \varphi^{\mathrm{Mix}}$	$\mathcal{A}_c := \mathcal{A}_c^{\mathrm{Oracle}}$
Case 2	$\varphi_c^{(0)} := 0$	$\mathcal{A}_c := \mathcal{A}_c^{\mathrm{Oracle}}$

Table 6.1. Configuration of the test cases in the experiment under laboratory conditions.

uncompressed PCM WAV format with 44.1 kHz sampling rate, 16 Bit, mono. Mixing all three single tracks together, we obtain 60 mixture signals. Additionally, the onset times and thus the approximate n_0 of all onsets are available per individual instrument. Using this information, we constructed a test set of 4421 drum onset events by taking excerpts from the mixtures, each located between consecutive onsets of the target instrument. In doing so, we zero pad N samples ahead of each excerpt. The rationale is to deliberately prepend a section of silence in front of the local transient position. Inside that section, decay influence of preceding note onsets can be ruled out and potentially occurring pre-echos can be measured. In turn, this leads to a virtual shift of the local transient location to $n_0 + N$ (which we denote again as n_0 for notational convenience). In Figure 6.3, the adjusted excerpt boundaries are visualized by the dashed blue lines and the virtually shifted n_0 by the blue line. Since the drum loops are realistic rhythms, the excerpts exhibit varying degree of superposition with the remaining drum instruments played simultaneously. In Figure 6.3a, the mixture (top) exhibits pronounced influence of the kick drum compared to the isolated hi-hat signal (bottom). For comparison, the two top plots in Figure 6.2a show a longer excerpt of the mixture x and the hi-hat component x_3 of our example signal. In the bottom plot in Figure 6.3a, one can see the kick drum x_1 in isolation. It is sampled from a Roland TR 808 drum computer and resembles a decaying sinusoid.

6.4.2 Evaluation Setting

For each mixture excerpt, we compute the STFT via (6.1) with $H = 512$ and $N = 2048$ and denote it as $\mathcal{X}^{\mathrm{Mix}}$. Since all test items have 44.1 kHz sampling rate, the frequency resolution is approx. 21.5 Hz and the temporal resolution is approx. 11.6 ms. We use a symmetric Hann window of size N for w. As a reference target, we take the same excerpt boundaries, apply the same zero-padding, but this time from the single track of each individual drum instrument, denoting the resulting STFT as $\mathcal{X}_c^{\mathrm{Oracle}}$. Subsequently, we define two different cases for the initialization of $\mathcal{X}_c^{(0)}$ as detailed in Table 6.1. Using these settings, we expect the inconsistency of the resulting $\mathcal{X}_c^{(0)}$ to be lower in case 1 compared to case 2. Knowing that there exists a consistent $\mathcal{X}_c^{\mathrm{Oracle}}$, we go through $L = 200$ iterations of both GL and our proposed TR method as described in Section 6.3.2.

6.4.3 Quality Measures

We introduce $\mathcal{G}\left(\mathcal{X}_c^{(\ell)}\right) := \mathrm{STFT}\left(\mathrm{iSTFT}\left(\mathcal{X}_c^{(\ell)}\right)\right)$ to denote successive application of the iSTFT and STFT (core of the GL algorithm) on $\mathcal{X}_c^{(\ell)}$. Following [127], we compute at each iteration ℓ the normalized consistency measure (NCM) as

$$\mathcal{C}\left(\mathcal{X}_c^{(\ell)}, \mathcal{X}_c^{\mathrm{Oracle}}\right) := 10\log_{10}\frac{\left\|\mathcal{G}\left(\mathcal{X}_c^{(\ell)}\right) - \mathcal{X}_c^{\mathrm{Oracle}}\right\|^2}{\left\|\mathcal{X}_c^{\mathrm{Oracle}}\right\|^2}, \tag{6.6}$$

for both test cases (see Table 6.1). As a more dedicated measure for the transient restoration, we compute the pre-echo energy as

$$\mathcal{E}\left(x_c^{(\ell)}\right) := \sum_{n=n_0-N}^{n_0}\left|x_c^{(\ell)}(n)\right|^2, \tag{6.7}$$

from the section between the excerpt start and the transient location in the intermediate, time-domain component signal reconstructions $x_c^{(\ell)} := \mathrm{iSTFT}\left(\mathcal{X}_c^{(\ell)}\right)$ for both test cases (see Table 6.1).

6.4.4 Results and Discussion

Figure 6.4 shows the evolution of both quality measures from (6.6) and (6.7) with respect to ℓ. Diagram 6.4(a) indicates that, on average, the proposed TR method performs equally well as GL in terms of inconsistency reduction. In both test cases, the curves for TR (solid line) and GL (dashed line) are almost indistinguishable, which indicates that our new approach shows similar convergence properties as the original method. As expected, the blue curves (Case 1) start at much lower initial inconsistency than the red curves (Case 2), which is clearly due to the initialization with the mixture phase φ^{Mix}. Diagram 6.4(b) shows the benefit of TR for pre-echo reduction. In both test cases, the pre-echo energy for TR (solid lines) is around 15 dB lower and shows a steeper decrease during the first few iterations compared to GL (dashed line). Again, the more consistent initial $\mathcal{X}_c^{(0)}$ of Case 1 (blue lines) exhibit a considerable head start in terms of pre-echo reduction compared to Case 2 (red lines). From these results, we infer that it is sufficient to apply only a few iterations (e.g., $L < 20$) of the proposed method in cases where reasonable initial phase and magnitude estimates are available. However, we need to apply more iterations (e.g., $L < 200$) in case we have a good magnitude estimate in conjunction with a weak phase estimate and vice versa. In the following, we will assess if our preliminary findings obtained under laboratory conditions hold true in a more realistic scenario.

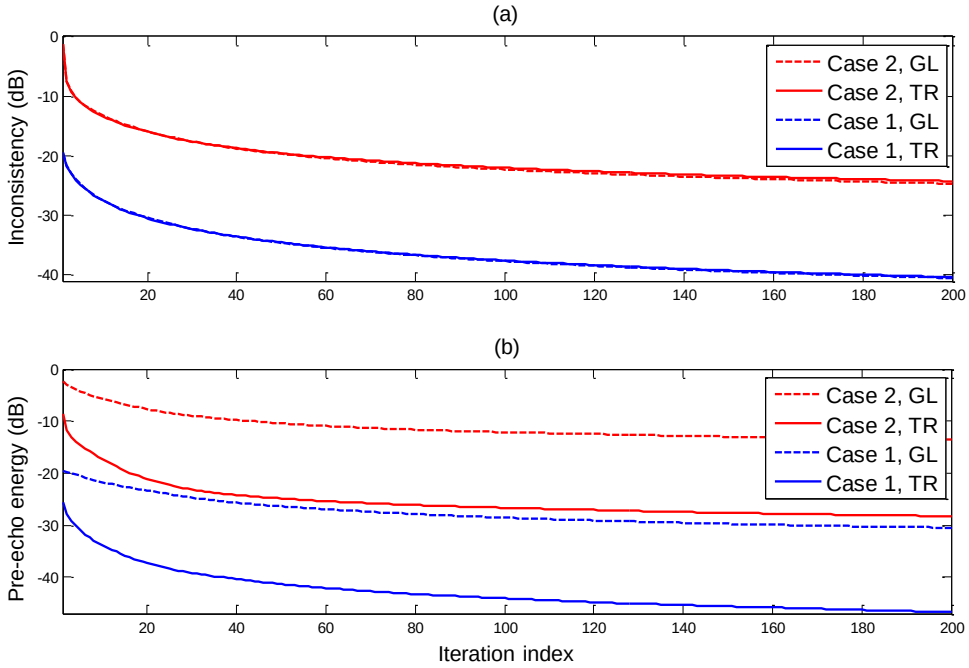

Figure 6.4. **(a)** Evolution of the normalized consistency measure vs. the number of iterations. **(b)** Evolution of the pre-echo energy vs. the number of iterations. The curves show the average over all test excerpts.

6.5 Application to NMF-based Audio Decomposition

In this section, we describe how to apply our proposed transient restoration method in a score-informed audio decomposition scenario. As in Section 6.4, our objective is again the extraction of isolated drum sounds from polyphonic drum recordings with enhanced transient preservation. In contrast to the idealized laboratory conditions we used before, we now estimate the magnitude spectrograms of the component signals from the mixture. To this end, we employ NMFD [133, 174, 185] as decomposition technique. We briefly describe our strategy to enforce score-informed constraints on NMFD. Finally, we repeat the experiments described Section 6.4 under these more realistic conditions and discuss our observations.

6.5.1 Spectrogram Decomposition via NMFD

In this section, we briefly review the NMFD method that we employ for decomposing the TF-representation of x. As indicated in Section 6.2.3, a wide variety of alternative separation approaches exists. Previous works [133, 174, 185] successfully applied NMFD, a convolutive version of NMF, for drum sound detection and separation. Intuitively speaking, the underlying, convolutive model assumes that all audio events in one of the component signals can be explained by a prototype event that acts as an impulse response to some onset-related activation (e.g.,

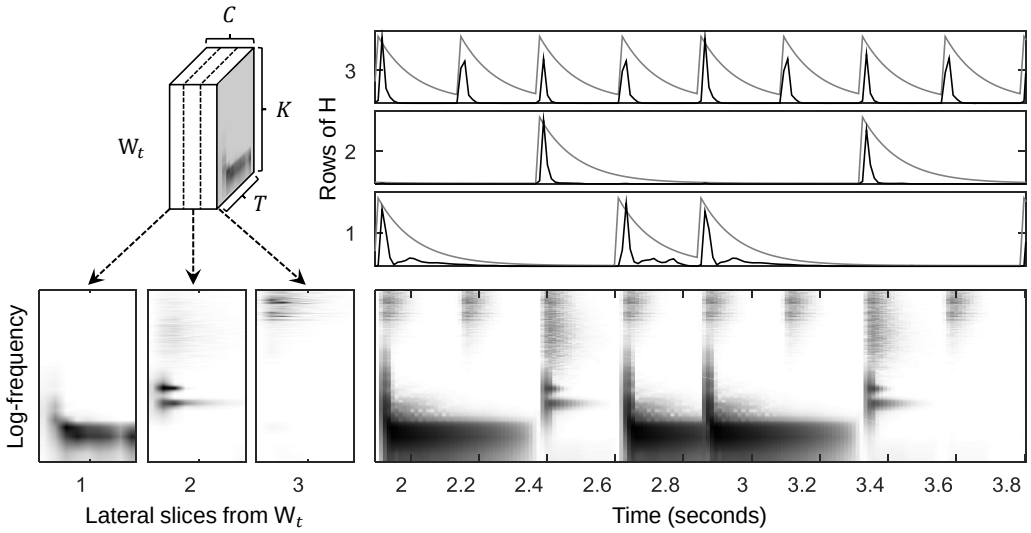

Figure 6.5. NMFD templates and activations computed for the example drum recording from Figure 6.2. The magnitude spectrogram V is shown in the lower right plot. The three leftmost plots are the spectrogram templates in W_t that have been extracted via NMFD. Their corresponding activations in H are shown as black curves in the three top plots. The gray curves show the score-informed initialization $\mathrm{H}^{(0)}$.

striking a particular drum). In Figure 6.2b, one can see this kind of behavior in the hi-hat component V_3. There, all instances of the 8 onset events look more or less like copies of each other that could be explained by inserting a prototype event at each onset position.

NMF can be used to compute a factorization $\mathrm{V} \approx \mathrm{W} \cdot \mathrm{H}$, where the columns of $\mathrm{W} \in \mathbb{R}_{\geq 0}^{K \times C}$ represent spectral basis functions (also called templates) and the rows of $\mathrm{H} \in \mathbb{R}_{\geq 0}^{C \times M}$ contain time-varying gains (also called activations). NMFD extends this model to the convolutive case by using two-dimensional templates so that each of the C spectral bases can be interpreted as a magnitude spectrogram snippet consisting of $T \ll M$ spectral frames. To this end, the convolutive spectrogram approximation $\mathrm{V} \approx \tilde{\mathrm{V}}$ is modeled as

$$\tilde{\mathrm{V}} := \sum_{t=0}^{T-1} \mathrm{W}_t \cdot \overset{t\rightarrow}{\mathrm{H}}, \tag{6.8}$$

where $\overset{t\rightarrow}{(\cdot)}$ denotes a frame shift operator. As before, each column in $\mathrm{W}_t \in \mathbb{R}_{\geq 0}^{K \times C}$ represents the spectral basis of a particular component, but this time we have T different versions of the component available. If we take lateral slices along selected columns of W_t, we can obtain C prototype magnitude spectrograms as depicted on the left hand side of Figure 6.5. NMFD typically starts with a suitable initialization of matrices $\mathrm{W}_t^{(0)}$ and $\mathrm{H}^{(0)}$. Subsequently, these matrices are iteratively updated to minimize a suitable distance measure between the convolutive approximation $\tilde{\mathrm{V}}$ and V. In this work, we use the update rules detailed in [185], which we omit for brevity.

6.5.2 Score-Informed NMFD

Proper initialization of $\mathrm{W}_t^{(0)}$ and $\mathrm{H}^{(0)}$ is an effective means to constrain the degrees of freedom in the NMFD iterations and enforce convergence to a desired, musically meaningful solution. One possibility is to impose score-informed constraints derived from a time-aligned, symbolic transcription [67]. To this end, the individual rows of $\mathrm{H}^{(0)}$ are initialized as follows: Each frame corresponding to an onset of the respective drum instrument is initialized with an impulse of unit amplitude, all remaining frames with a small constant. Afterward, we apply a nonlinear exponential moving average filter to model the typical short decay of a drum event. The outcome of this initialization is shown in the top three plots of Figure 6.5 (gray curves).

In [66], best separation results were obtained by score-informed initialization of both the templates and the activations. For separation of pitched instruments (e.g., piano), prototypical overtone series can be constructed in $\mathrm{W}_t^{(0)}$. For drums, it is more difficult to model prototype spectral bases. Thus, it has been proposed to initialize the bases with averaged or factorized spectrograms of isolated drum sounds [7, 37, 133]. In this chapter, we use a simple alternative that first computes a conventional NMF whose activations H and templates W are initialized by the score-informed $\mathrm{H}^{(0)}$ and setting $\mathrm{W}^{(0)} := 1$.

With these settings, the resulting factorization templates are usually a pretty decent approximation of the average spectrum of each involved drum instrument. Simply replicating these spectra for all $t \in [0 : T-1]$ serves as a good initialization for the template spectrograms. After some NMFD iterations, each template spectrogram typically corresponds to the prototype spectrogram of the corresponding drum instruments and each activation function corresponds to the deconvolved activation of all occurrences of that particular drum instrument throughout the recording. A typical decomposition result is shown in Figure 6.5, where one can see that the extracted templates (three leftmost plots) indeed resemble prototype versions of the onset events in V (lower right plot). Furthermore, the location of the impulses in the extracted H (three topmost plots) are very close to the maxima of the score-informed initialization.

In the following, we describe how to further process the NMFD results in order to extract the desired components. Let $\mathrm{H} \in \mathbb{R}_{\geq 0}^{C \times M}$ be the activation matrix learned by NMFD. Then, we define for each $c \in [1 : C]$ the matrix $\mathrm{H}_c \in \mathbb{R}_{\geq 0}^{C \times M}$ by setting all elements to zero except for the c^{th} row that contains the desired activations previously found via NMFD. We approximate the c^{th} component magnitude spectrogram by $\tilde{\mathrm{V}}_c := \sum_{t=0}^{T-1} \mathrm{W}_t \cdot \overset{t\rightarrow}{\mathrm{H}_c}$.

Since the NMFD model yields only a low-rank approximation of V, spectral nuances may not be captured well. In order to remedy this problem, it is common practice to calculate soft masks that can be interpreted as a weighting matrix reflecting the contribution of $\tilde{\mathrm{V}}_c$ to the mixture V. The mask corresponding to the desired component can be computed as $\mathrm{M}_c := \tilde{\mathrm{V}}_c \oslash \left(\epsilon + \sum_{c=1}^{C} \tilde{\mathrm{V}}_c\right)$, where $\oslash$ denotes element-wise division and ϵ is a small positive constant to avoid division by zero. We obtain the masking-based estimate of the component magnitude spectrogram as $\mathrm{V}_c := \mathrm{V} \odot \mathrm{M}_c$,

Test case	Initial phase estimate	Fixed magnitude estimate
Case 3	$\varphi_c^{(0)} := \varphi^{\mathrm{Mix}}$	$\mathcal{A}_c := \mathrm{V}_c^{\top}$
Case 4	$\varphi_c^{(0)} := 0$	$\mathcal{A}_c := \mathrm{V}_c^{\top}$

Table 6.2. Configuration of the test cases in the second experiment involving score-informed audio decomposition.

with $\odot$ denoting element-wise multiplication. This procedure is referred to as α-Wiener filtering in [134].

6.5.3 Evaluation Results

We now basically repeat the experiment from Section 6.4, keeping the STFT parameters and excerpt boundaries as described in Section 6.4.2. The component magnitude spectrograms are estimated from the mixture using $L^{\mathrm{NMF}} = 30$ NMFD iterations and spectrogram templates with a duration of $T = 8$ frames (approx. 100 ms). Consequently, we introduce two new test cases as detailed in Table 6.2.

In Figure 6.6a, we again observe that the inconsistency reduction obtained using TR reconstruction (solid lines) is indistinguishable from the GL method (dashed lines). The improvements are less significant compared to the numbers that can be obtained when using oracle magnitude estimates (compare Figure 6.4a). On average, the reconstructions in Case 3 (initialized with φ^{Mix}) seem to quickly get stuck in a local optimum. Presumably, this is due to imperfect NMFD decomposition of the onset related spectrogram frames, where all instruments exhibit a more or less flat magnitude distribution and thus show increased spectral overlap.

In Figure 6.6b, we first see that pre-echo reduction with NMFD-based magnitude estimates $\mathcal{A}_c := \mathrm{V}_c^{\top}$ and zero phase (Case 4) works slightly worse than in Case 2 (compare Figure 6.4b). This supports our earlier findings, that weak initial phase estimates benefit the most from applying many iterations of the proposed method. GL reconstruction using φ^{Mix} (Case 3) slightly increases the pre-echno energy over the iterations. In contrast, applying the TR reconstruction decreases the pre-echo energy by roughly -3 dB, which amounts to approx. 15 % of the improvement achievable under idealized conditions (Case 1).

In Figure 6.3, different reconstructions of a selected hi-hat onset from our example drum loop is shown in detail. Regardless of the used magnitude estimate (oracle in (b) or NMFD-based in (c)), the proposed TR reconstruction (bottom) clearly exhibits reduced pre-echos in comparison to the conventional GL reconstruction (top). We provide example component signals from this drum loop and a few test items online[29]. By informal listening (preferably using headphones), one can

[29] Audio examples: `http://www.audiolabs-erlangen.de/resources/MIR/2015-DAFx-TransientRestoration/`, last accessed June 14, 2018

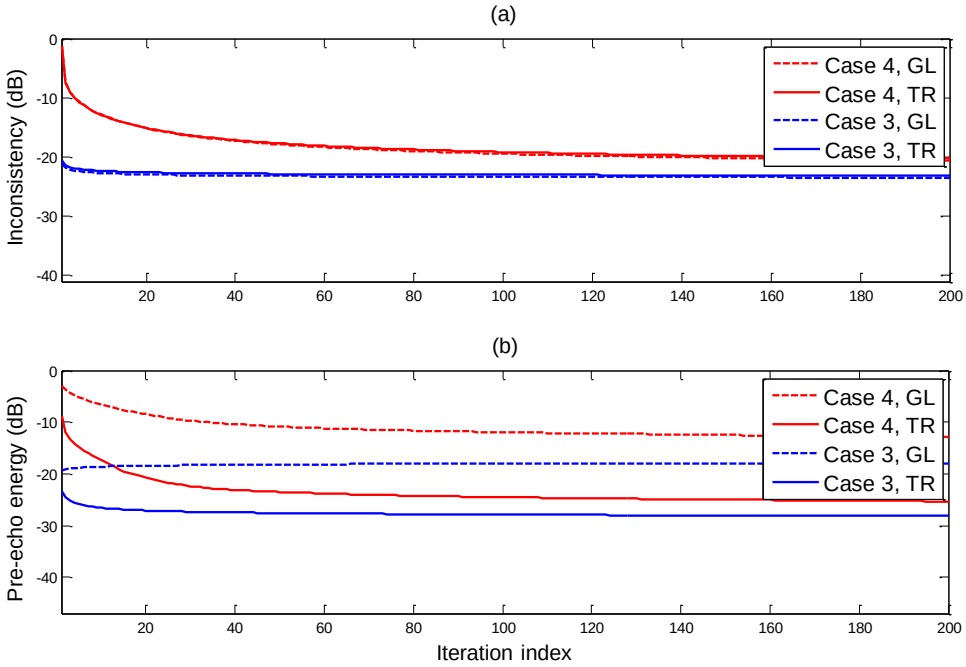

Figure 6.6. **(a)** Evolution of the normalized consistency measure vs. the number of iterations. **(b)** Evolution of the pre-echo energy vs. the number of iterations. The curves show the average over all test excerpts, the axis limits are the same as in Figure 6.4.

clearly spot differences in the onset clarity that can be achieved with different combinations of MSTFT initializations and reconstruction methods. Even in cases, where imperfect magnitude decomposition leads to undesired cross-talk artifacts in the single component signals, our proposed TR method better preserves transient characteristics than the conventional GL reconstruction. Furthermore, usage of the mixture phase for MSTFT initialization seems to be a good choice since one can often notice subtle differences in the reconstruction of the drum events' decay phase in comparison to the oracle signals. However, timbre differences caused by imperfect magnitude decomposition are much more pronounced.

6.6 Conclusions and Further Notes

In this chapter, we proposed a simple, yet effective extension to Griffin and Lim's iterative LSEE-MSTFTM procedure (GL) for improved restoration of transient signal components in music source separation. The method requires additional side information about the location of the transients, which we assume as given in an informed source separation scenario. Two experiments with the publicly available "IDMT-SMT-Drums" dataset showed that our method is beneficial for reducing pre-echos both under laboratory conditions as well as for component signals obtained using a state-of-the-art source separation technique. Future work will be directed

towards automatic estimation of the required transient positions and application of this technique for polyphonic music recordings involving more than just drums and percussion.

Chapter 7

Generalized Wiener Filtering and Kernel Additive Modeling

The work in this chapter is mainly based on our contribution in [46].

Music source separation aims at decomposing music recordings into their constituent component signals. Many existing techniques are based on separating a time-frequency representation of the mixture signal by applying suitable modeling techniques in conjunction with generalized Wiener filtering. Recently, the term α-Wiener filtering was coined together with a theoretic foundation for the long-practiced use of magnitude spectrogram estimates in Wiener filtering. So far, optimal values for the magnitude exponent α have been empirically found in oracle experiments regarding the additivity of spectral magnitudes. In the first part of this chapter, we extend these previous studies by examining further factors that affect the choice of α. In the second part, we investigate the role of α in Kernel Additive Modeling applied to harmonic-percussive source separation. Our results indicate that the parameter α may be understood as a kind of selectivity parameter, which should be chosen in a signal-adaptive fashion.

7.1 Introduction

Music signals can be understood as superposition (mixture) of different sound sources (components), such as melodic instruments, singing voice, bass, and drums. Music source separation aims at recovering these constituent component signals from the mixture. The separated sources may be used for music retrieval tasks, automatic music transcription, as well as music production and restoration, see [67] for an overview.
At the core, many source separation techniques try to extract the target component signal from

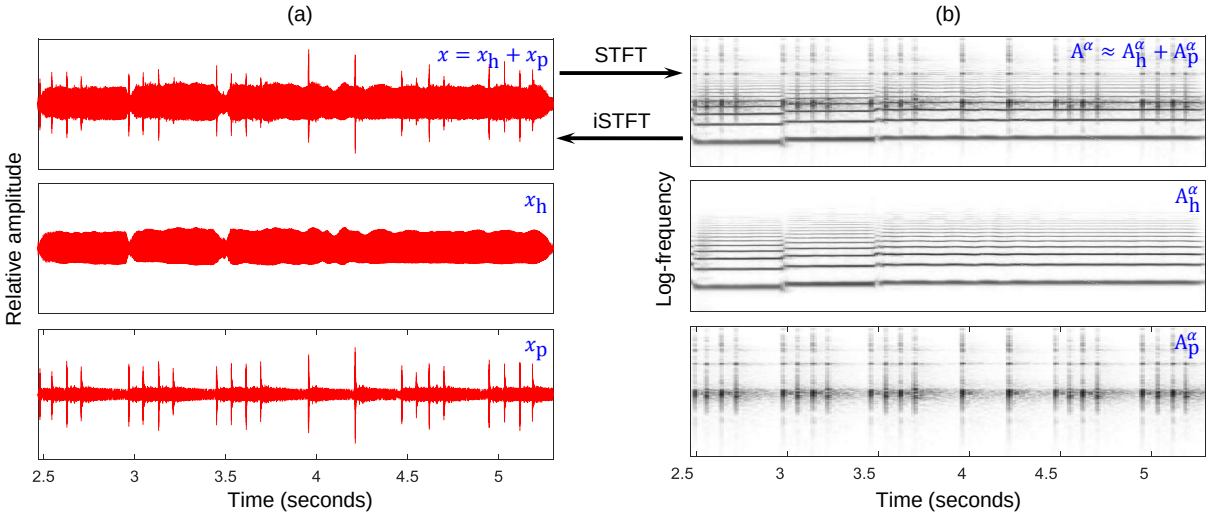

Figure 7.1. Illustration of our signal model. **(a)** Mixture signal $x := x_\mathrm{h} + x_\mathrm{p}$, which is the superposition of two source signals x_h and x_p. **(b)** Magnitude spectrogram of the mixture A and the sources A_h and A_p. For better visibility, we use a logarithmically spaced frequency axis and logarithmic magnitude compression.

the mixture by means of time-variant filtering. In practice, this filtering procedure is commonly realized by element-wise weighting of the mixture's short-time Fourier transform (STFT) with some kind of time-frequency (TF) mask. Besides the wide-spread use of *binary masks* [216], many approaches use so-called *soft masks*. The most common strategy to construct soft masks is to use generalized Wiener filtering [11, 72, 134]. Loosely speaking, this procedure consists of first estimating the spectrogram of the target source and subsequently taking its ratio to the sum of all source spectrogram estimates as the filter weight. In order to disambiguate the usage of the term spectrogram throughout the literature, we use the notion of an α-spectrogram as introduced in [134], meaning the modulus of the STFT raised to some arbitrary exponent $\alpha \in [0, 2]$. With this clarification, Wiener filtering relies on the rather strong assumption that the sources' α-spectrograms add up to the mixture's α-spectrogram. This completely neglects possible phase-related issues, such as destructive interference [86]. While some research effort has been dedicated to incorporating phase information [86, 117, 129], other authors have attempted to find more appropriate masking strategies. In [72], an alternative family of TF masks based on well-known divergence measures such as the Kullback-Leibler and Itakura-Saito was proposed. In a similar fashion, [120] tried to find an optimal magnitude exponent (among other parameters) in diverse source separation tasks. *Oracle* source separation experiments with known component signals were conducted in [215] in order to identify a domain satisfying the additivity assumption of spectral magnitudes. Similar settings were used in [134], where the authors also established a theoretic foundation for using magnitude instead of power spectra (i.e., $\alpha = 1$ instead of $\alpha = 2$) in Wiener filtering.

From the literature, we see that the magnitude exponent α is considered to be an important

parameter, which is not fully understood yet. In this chapter, we take a different perspective and investigate two aspects of α in an experimental fashion. In Section 7.2, we extend the oracle experiments from [134] in order to assess the dependency of the additivity assumption of α-spectrograms on the signal type as well as other influential factors. In Section 7.3, we asses the influence of α in Kernel Additive Modeling (KAM), a recently proposed [135] source separation procedure that strongly relies on iteratively applying Wiener filtering. As we will show, α can be understood as a selectivity parameter to trade off between interference reduction and artifacts depending on the target signal type.

7.2 Additivity of α-Spectrograms Revisited

In this section, we present the settings and results of oracle experiments on the additivity assumption of α-spectrograms. As in related studies [134, 215], we work with known source signals in order to create a controlled scenario that is independent of specific source separation methods. Our initial question is if it is beneficial to choose α differently depending on whether we want to separate a saxophone solo from accompanying instruments or if we want to separate drum instruments from a drum solo recording.

7.2.1 Notation and Signal Model

Let $x : \mathbb{Z} \to \mathbb{R}$ be the real-valued, discrete-time domain mixture signal that is based on the linear superposition $x := x_\mathrm{h} + x_\mathrm{p}$ of two component signals corresponding to the individual sources. Example component signals are illustrated in Figure 7.1a, where x_h is a harmonic melody instrument and x_p is a percussive accompaniment. We will return to these specific music signal properties in Section 7.3. As already discussed, we transition to the TF domain as depicted Figure 7.1b. To this end, let $\mathrm{A}\,(k, m)$ be the non-negative modulus of the STFT at the k^{th} spectral bin and the m^{th} time frame. As shown in the top plot of Figure 7.1b, we assume that the α-spectrograms A_h and A_p approximately add up to the mixture:

$$\mathrm{A}^\alpha \approx \mathrm{A}_\mathrm{h}^\alpha + \mathrm{A}_\mathrm{p}^\alpha. \tag{7.1}$$

Here, the magnitude exponent $\alpha \in [0, 2]$ is applied in an element-wise fashion. Although our notation easily extends to the more general scenario involving an arbitrary number $C \in \mathbb{N}$ of sources, we will restrict ourselves to the case $C = 2$ in the following discussion for the sake of clarity.

7.2.2 Signal-Dependency of α

First, we repeat a similar experiment as in [134], using multi-track recordings to quantify the additivity of α-spectrograms under varying α. The basic protocol is to create linear mixtures from oracle source signals and then switch to the TF domain to assess the additivity assumption. With respect to our running example (see Figure 7.1), this can be formalized as computing a suitable divergence $\mathcal{D}$ between the mixture's α-spectrogram and the sum of the sources' α-spectrograms as

$$\mathcal{D}\left(\mathrm{A}^{\alpha}, \mathrm{A}_1^{\alpha} + \mathrm{A}_2^{\alpha}\right) \text{ for } \alpha \in [0.2, 2]\,. \tag{7.2}$$

As in [134], the metric $\mathcal{D}$ can be either the α-dispersion, Itakura-Saito, or Kullback-Leibler divergence. In our experiments, we use source signals from the "QUASI"[30] dataset. This set consists of several full-length songs from diverse music genres, each providing single track recordings of the involved sources, such as singing voice, melodic instruments, bass, drums, or percussion. Thus, the set covers a broad range of different signals characteristics, in the sense that it contains harmonic as well as percussive instruments with varying degree of interdependence between them.

In particular, we are interested if the results reported in [134] generalize to other, more homogeneous types of music recordings. Thus, we extend the experiment with two additional datasets. The first consists of all single tracks of the "Bach10"[31] and "TRIOS"[32] corpora, which are dominated by harmonic instruments, such as violin, viola, bassoon, horn, clarinet, and piano. It should be noted that one piece in the TRIOS corpus contains three drum tracks, these have been excluded for our experiment. The second set uses drum only recordings from the "IDMT-SMT-Drums" dataset[33], where the sources correspond to the three drum instruments kick drum, snare drum, and hi-hat [37]. Across all sets, the audio items are in uncompressed PCM WAV format with 44.1 kHz sampling rate, 16 Bit, mono. As for the STFT parameters, we adopt the settings from [134], using Hamming-windowed frames of approx. 80 ms duration and 80% overlap between them.

As is shown in Figure 7.2b we can not exactly replicate the curves reported in [134] (dashed lines) but the tendencies are similar. However, we can indeed see a different behavior of the curves in Figure 7.2a and Figure 7.2c. Besides different minimum positions, it is remarkable that the curves in (c) are much flatter. As a tendency, one might say that for drums, the range of possible quasi-optimal α is much broader than with harmonic instruments. As we will show

[30] `http://www.tsi.telecom-paristech.fr/aao/en/2012/03/12/quasi/`, last accessed June 14, 2018
[31] `http://www.ece.rochester.edu/~zduan/resource/Resources.html`, last accessed June 14, 2018
[32] `http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/27`, last accessed June 14, 2018
[33] `http://www.idmt.fraunhofer.de/en/business_units/smt/drums.html`, last accessed June 14, 2018

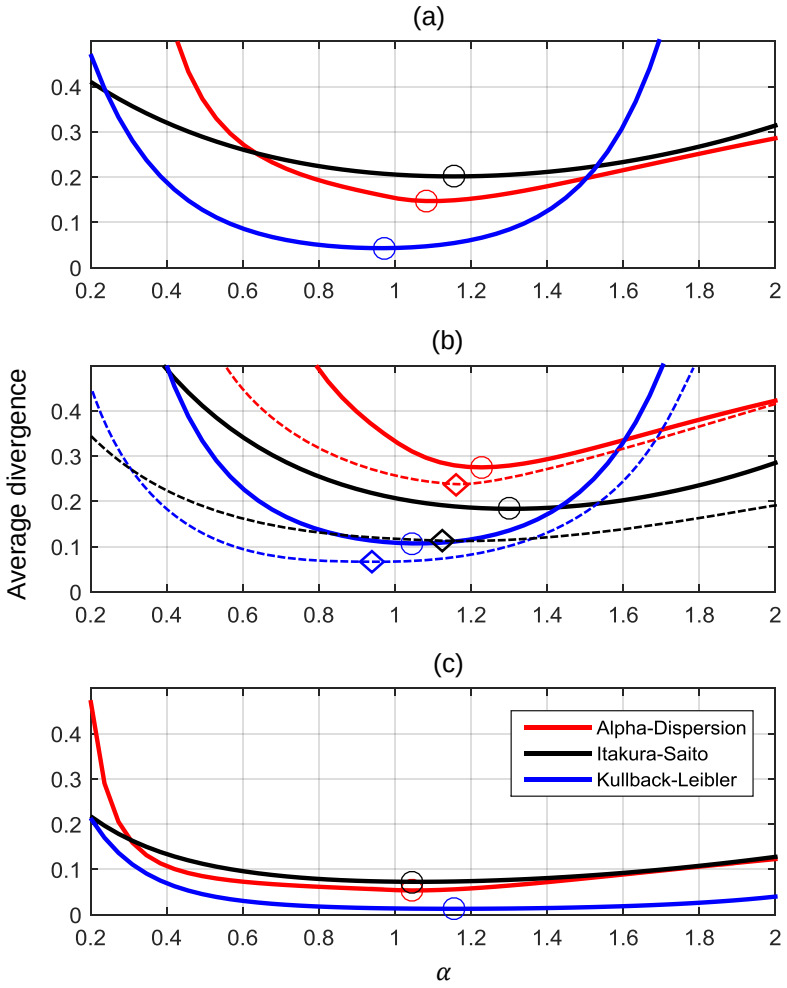

Figure 7.2. Average α-dispersion, Itakura-Saito and Kullback-Leibler divergences as a function of α. Global minimum positions are are marked with a circle. The legend in **(c)** applies to all plots. **(a)** Results obtained with purely harmonic sources (Bach10 & TRIOS datasets). **(b)** Results obtained with harmonic and percussive sources (QUASI dataset). The dashed lines provide the original results from [134] for comparison, diamond markers represent the respective minimum positions. **(c)** Results obtained with purely percussive sources (IDMT-SMT-DRUMS dataset).

in the next section, these results should be read with great care, since there are more factors involved than just the signal types.

7.2.3 Level-Dependency of α

In this second experiment, we basically repeat the same protocol as before. This time, the only difference is that each source signal is normalized so that its absolute maximum value is 1.0 before adding them up to the linear mixture. Since the normalization factor depends on the

properties of the respective signal, the normalization step is expressed by introducing modified sources signals $\overline{x}_{\mathrm{h}}$ and $\overline{x}_{\mathrm{p}}$ in

$$x := \overline{x}_{\mathrm{h}} + \overline{x}_{\mathrm{p}}. \tag{7.3}$$

As can be seen in Figure 7.3, this simple modification affects the results quite a lot. As expected, the Itakura-Saito divergences (black curves) stay the same for all datasets, since they are less susceptible to level differences. The α-dispersion (red curves) look like scaled versions of the ones in Figure 7.2, a direct consequence of its calculation rule given in [134]. For the QUASI dataset, the scaling is so pronounced that the curve is out the plot range in Figure 7.3. However, at least the minimizing α-values stay the same. In contrast, the Kullback-Leibler divergences (blue curves) look completely different and the minimizing α-values end up in different positions, so the choice of an optimal α for a certain signal type becomes questionable. From the empirical results in [134], one could get the impression that an $\alpha \approx 1.2$ is a sensible choice for general purpose music source separation (see Figure 7.2). Our results rather indicate that the choice of α is sensitive to a number of additional factors. This is in line with the findings in [215], where also the number of sources C has been shown to be influential. Other factors, such as the mutual correlation between the sources remain completely nebulous and should be addressed in further studies.

7.3 Influence of α in Kernel Additive Modeling

After these oracle-based experiments, we now want to study the influence of α in a concrete source separation scenario. In particular, we consider the task of harmonic-percussive source separation (HPSS) as a specific case study. HPSS aims at splitting a music recording into harmonic (e.g., melodic instruments, tonal components) and percussive (e.g., drums and percussion, transient components) sources, see Figure 7.1 for an example. A high quality HPSS is an important pre-requisite for advanced tasks such drum sound separation [37, 39].
A comprehensive overview of recent methods for HPSS is given in [199]. Many works already discussed the problem that music recordings may consist of sounds that are neither clearly harmonic nor percussive [84, 155]. An example are harmonic tones, whose fundamental frequency is modulated over time as is typical in recordings of instruments with vibrato. We neglect this problem in our study for the sake of compactness.

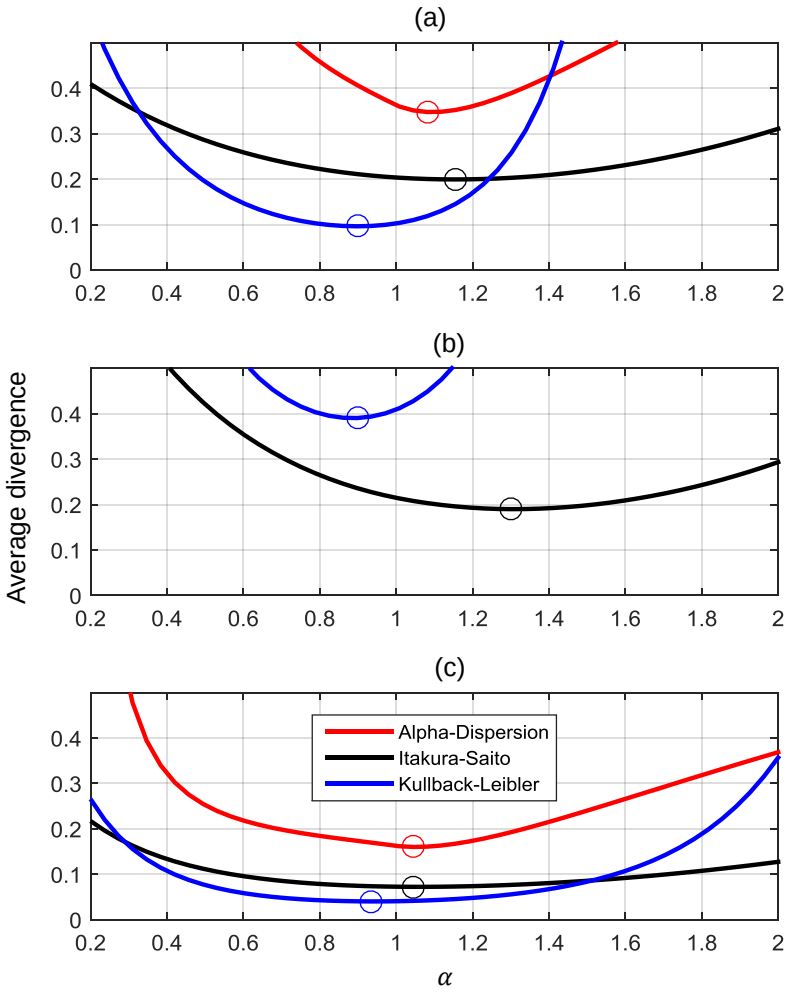

Figure 7.3. Three different divergences are shown as a function of α, the same description applies as in Figure 7.2. **(a)** Normalized Bach10 & TRIOS sources. **(b)** Normalized QUASI sources. The curve for α-dispersion is outside the plot range. **(c)** Results from normalized IDMT-SMT-DRUMS sources.

7.3.1 KAM-Based HPSS

Recently, a novel class of source separation approaches was proposed under the notion of Kernel Additive Modeling (KAM) [135, 165]. In contrast to global decomposition paradigms (e. g., Non-negative Matrix Factorization), KAM exploits local regularities of the source spectrograms. An HPSS variant based on KAM was introduced in [78], which is also the main technique used in our study. The method relies on iteratively modeling the component spectrograms and applying Wiener filtering as depicted in Figure 7.4. With respect to our two-component signal model, the α-Wiener masks for the two components are computed as

$$\mathrm{M_h} := \mathrm{B_h^\alpha} \oslash \left(\mathrm{B_h^\alpha} + \mathrm{B_p^\alpha}\right) \quad \text{and} \quad \mathrm{M_p} := \mathrm{B_p^\alpha} \oslash \left(\mathrm{B_h^\alpha} + \mathrm{B_p^\alpha}\right), \tag{7.4}$$

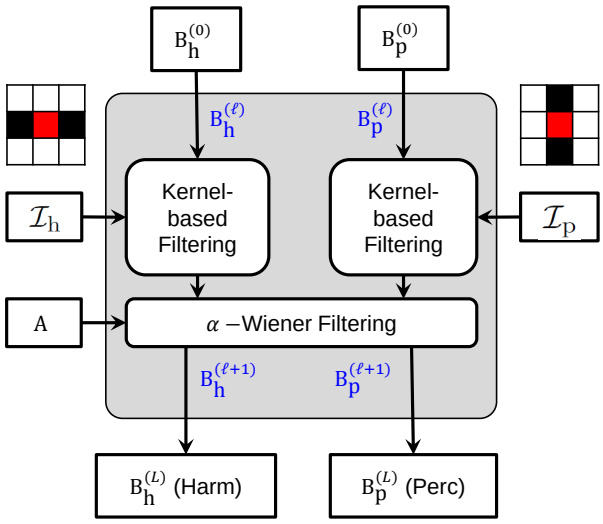

Figure 7.4. Overview of the KAM-based algorithm for HPSS. The gray box stands for iterative refinement.

where $\oslash$ denotes element-wise division, $\mathrm{B_h}$ represents the harmonic and $\mathrm{B_p}$ the percussive component estimate.

In our reimplementation of KAM-based HPSS, we set the initial estimate $\mathrm{B_h^{(0)}}$ and $\mathrm{B_p^{(0)}}$ to the mixture α-spectrogram A^α. As in [78], we construct two kernels $\mathcal{I}_\mathrm{h}$ and $\mathcal{I}_\mathrm{p}$ for the enhancement of harmonic and percussive structures. As illustrated by Figure 7.4, the harmonic kernel is all zero except one horizontal row and the percussive kernel shows a perpendicular structure. We introduce the iteration index $\ell = 0, 1, 2, \ldots, L \in \mathbb{N}$ and proceed with iterative refinements by first applying a kernel-based filtering to each of the components and subsequently applying (7.4). It should be noted that the original procedure in [78] applies 2D median filters in the kernel-based filtering stage. In contrast, we use 2D convolution with the kernels $\mathcal{I}_\mathrm{h}$ and $\mathcal{I}_\mathrm{p}$. The rationale behind replacing median filtering by convolution is that we want to eliminate any other nonlinear operations besides raising the STFT modulus to the magnitude exponent α in (7.4).

7.3.2 Evaluation in HPSS Task

To investigate the influence of α-Wiener filtering in KAM-based HPSS, we generate one test item by superimposing a real-world trumpet melody (harmonic) with castanets (percussive). The experimental settings are the same as in [78], using an STFT blocksize of 4096 samples (approx. 92 ms) and a hopsize of 1024 samples (75% overlap). Kernels of 17×17 elements are used for KAM, and the number of iterations is set to $L = 10$. At each iteration, we evaluate the separation quality of the latest harmonic component estimate $\mathrm{B_h^{(\ell+1)}}$ and of the latest percussive component estimate $\mathrm{B_p^{(\ell+1)}}$, both of which are obtained after α-Wiener filtering. To this end, we use the mixture's phase spectrogram and apply inverse STFT to yield the time-domain reconstructions $\hat{x}_\mathrm{h}$ and $\hat{x}_\mathrm{p}$ respectively. In accordance to the standards used in the literature on music source

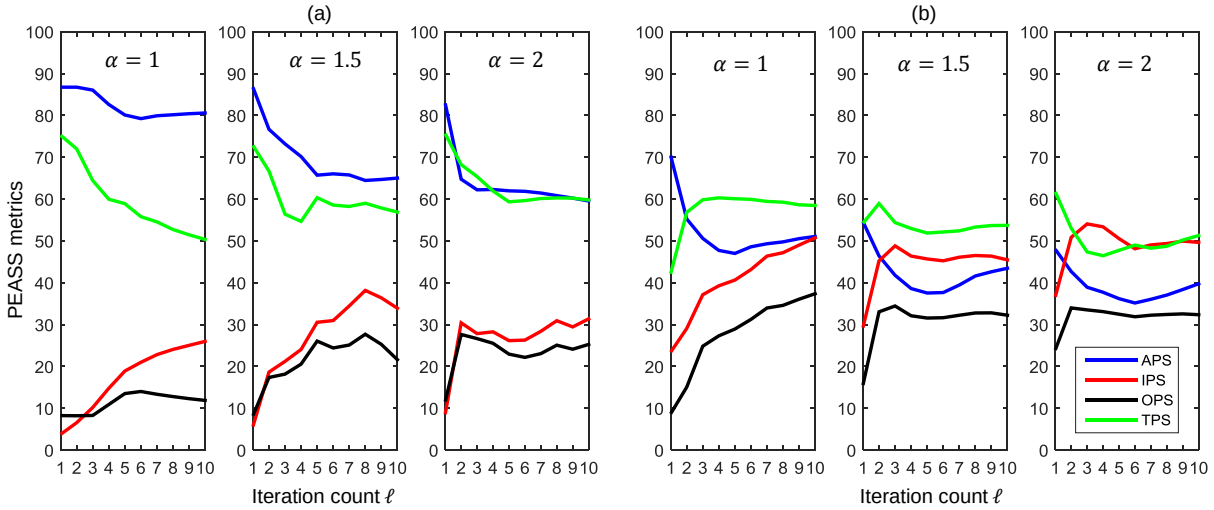

Figure 7.5. Evolution of the PEASS measures plotted along the iteration count ℓ. **(a)** Results for the harmonic component signal. **(b)** Results for the percussive component signal.

separation, we employ the Perceptual Evaluation Methods for Audio Source Separation (PEASS) [62, 208] in order to evaluate the quality of the reconstructed component signals. In contrast to the experiments in the original paper, we vary the α-parameter in (7.4) in order to assess its influence on this procedure.

In Figure 7.5, we show the evolution of four perceptually motivated PEASS metrics with increasing iteration count ℓ for the harmonic component in (a) and the percussive component in (b). The metrics comprise the artifact-related (APS), interference-related (IPS), target-related (TPS) and overall perceptual score (OPS), which can attain a maximal score of 100 in case of a perfect separation. In Figure 7.5a, it can be seen that the OPS benefits from higher α but quickly saturates already after a few iterations. As expected, the artifacts-related APS drops with increasing ℓ, while the interference-related IPS improves. In Figure 7.5b, the percussive component attains much lower APS even for $\alpha = 1$, probably due to pre-echos that occur when using large STFT blocksizes in conjunction with the mixture phase for reconstruction of transient signal components [38]. Informal listening tests confirmed that pre-echos are indeed an issue. However, it is interesting that the OPS and interference-related IPS can go much higher than for the harmonic component, and seem to be less susceptible to changing α. This indicates that it might be beneficial to use $\alpha \approx 1$ if we are interested in extracting the percussive component and $\alpha \approx 2$ for the harmonic component. We also want to stress how easily the PEASS scores can be increased or decreased by just changing α. In competitive evaluation campaigns such as SiSec[34], often a few score points can decide over the ranking of submitted source separation algorithms. The susceptibility to such basic parameters as the magnitude exponent α sheds a new light on the interpretation of these competition results.

[34] `https://sisec.inria.fr/`, last accessed June 14, 2018

7.4 Conclusions and Further Notes

In this chapter we reported results of exploratory experiments on the influence of the magnitude exponent α with respect to the additivity assumption as well as Wiener filtering in KAM-based HPSS. Empirically, we show that the choice of α is sensible to several factors, such as the signal types, the relative mixing levels and the number of sources. In KAM-based HPSS, where α-Wiener Filtering is applied iteratively, there is a delicate trade-off between softer separation with $\alpha = 1$ compared to stronger selectivity and faster convergence at the cost of undesired artifacts for $\alpha = 2$. Future work will be directed towards developing strategies for α-Wiener filtering that are adaptive to signal types, temporal evolution or even local spectral characteristics of the target sources.

Chapter 8

Harmonic-Percussive Source Separation

The work in this chapter is mainly based on the manuscript for [47].

This chapter addresses the separation of drums from music recordings, a task closely related to harmonic-percussive source separation (HPSS). In previous works, two families of algorithms have been prominently applied to this problem. They are based either on local filtering and diffusion schemes, or on global low-rank models. In this chapter, we propose to combine the advantages of both paradigms. To this end, we use a local approach based on Kernel Additive Modeling (KAM) to extract an initial guess for the percussive and harmonic parts. Subsequently, we use Non-Negative Matrix Factorization (NMF) with soft activation constraints as a global approach to jointly enhance both estimates. As an additional contribution, we introduce a novel constraint for enhancing percussive activations and a scheme for estimating the percussive weight of NMF components. Throughout the chapter, we use a real-world music example to illustrate the ideas behind our proposed method. Finally, we report promising BSS Eval results achieved with the publicly available test corpora ENST-Drums and QUASI, which contain isolated drum and accompaniment tracks.

8.1 Introduction

The general goal of music source separation is to decompose a recording into its constituent signal components. Considering the challenging scenario of percussive and inharmonic sound sources, we aim to extract drum sound events in a perceptually convincing quality that allows remixing and repurposing [39]. Going beyond our Chapter 4, we focus on drum recordings with

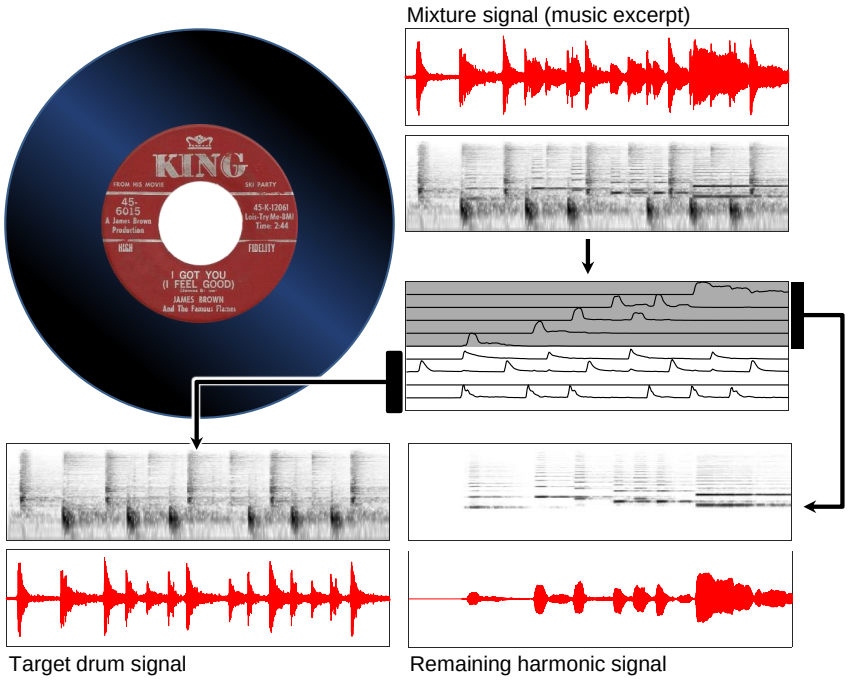

Figure 8.1. Illustration of our application scenario, the decomposition of a music recording into drums and remaining instruments.

moderate interference from melodic instruments (see Figure 8.1). In particular, we are interested in decomposing break sections that appear in pop, jazz, soul, and funk recordings of the 1960's to 1980's [138]. Such instrumental passages are often characterized by a pronounced drum beat interspersed with melodic instruments (e. g., bass, guitar, organ, saxophone, trumpet) and, rarely, singing voice.

In Figure 8.2, we introduce an idealized example for our source separation task using an excerpt of the 1964 recording of "I Got You (I Feel Good)," by James Brown & The Famous Flames. This short instrumental break features Maceo Parker playing alto sax over a four-bar drum beat played by his brother Melvin Parker. This running example appears several times throughout the chapter. For now, it is sufficient to realize that we are interested in removing the sound events of the alto sax (corresponding to the notes in the upper staff of Figure 8.2a).

Harmonic-percussive source separation (HPSS) seems like a natural choice for attenuating the interference of melodic instruments into the drum part. State-of-the-art HPSS methods are compared in detail in [199] and in [156]. Generally, there are two conceptually different HPSS approaches: local and global methods. Local HPSS methods emphasize localized time-frequency (TF) characteristics that distinguish drums from melodic instruments. In Section 8.2.1, we will briefly recapitulate a local HPPS method based on Kernel Additive Modeling (KAM) [46, 78]. Global HPSS methods decompose the mixture spectrogram into low-rank components according to a global optimization criterion. As we explain in Section 8.2.2, we use Non-Negative Matrix

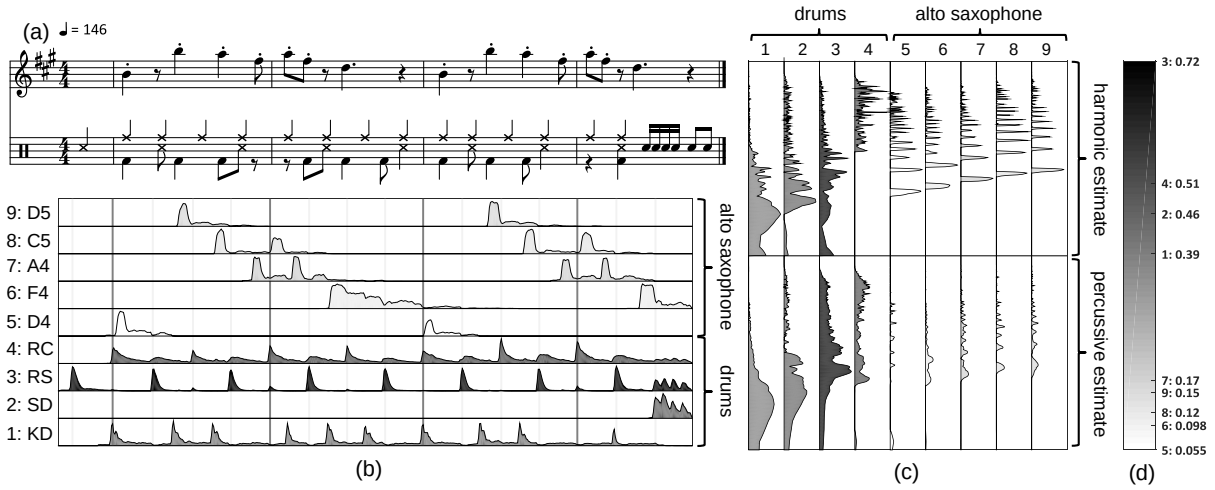

Figure 8.2. Instrumental break from "I Got You (I Feel Good)". **(a)** Score notation of alto sax and drums. **(b)** NMF activations of alto sax and drums. Bar and beat grid are indicated in the background and aligned to the score notation. **(c)** NMF templates corresponding to the activations. **(d)** Percussive weight $\mathrm{p}(r)$ encoded in the gray scale, with indices referring to the NMF components.

Factorization (NMF) to do so. Especially for HPSS, many authors advocate to apply constraints to NMF [24, 25, 54, 155, 156], exploiting the different TF characteristics of drums and melodic instruments.

In this chapter, we propose a novel two-stage approach, unifying local and global methods. In a first stage, KAM finds initial estimates for the percussive and harmonic parts (see Algorithm 2 in Section 8.2.1). The second stage then jointly refines these estimates using NMF with soft activation constraints (see Algorithm 3 in Section 8.2.4). As an additional contribution, we introduce the notion of percussive weight, an adaptive measure implicitly classifying NMF components according to their contribution to the drums. We explain how the percussive weight can be easily derived in our framework. Furthermore, we introduce a soft constraint for NMF activations that emphasizes drum-specific temporal characteristics. The experiments in Section 8.3 show that our proposed method yields improved BSS Eval measures on the ENST-Drums and QUASI corpora. Finally, in Section 8.4, we discuss strengths and weaknesses of our approach and point out remaining challenges.

8.2 Proposed System

As is common in music source separation, we decompose the mixture signal in the Short-time Fourier Transform (STFT) domain. To this end, let $\mathrm{A}(k, m)$ be the non-negative STFT magnitude at the k^{th} frequency bin and the m^{th} time frame, with $k \in [0 : K-1]$ and $m \in [0 : M-1]$.

The number of available bins $K \in \mathbb{N}$ and frames $M \in \mathbb{N}$ determines the dimension of our mixture spectrogram matrix $\mathrm{A} \in \mathbb{R}_{\geq 0}^{K \times M}$. Our objective is to split the mixture A in two complementary magnitude spectrograms $\mathrm{A_p}$ (drum part) and $\mathrm{A_h}$ (melodic part), such that $\mathrm{A} = \mathrm{A_p} + \mathrm{A_h}$. As shown in Figure 8.3, our main idea is to decompose A using KAM and NMF in a cascade. KAM serves to find initial estimates $\mathrm{A_p^{KAM}}$ and $\mathrm{A_h^{KAM}}$ which are then jointly refined by NMF, yielding the final $\mathrm{A_p^{NMF}}$ and $\mathrm{A_h^{NMF}}$.

8.2.1 Local HPSS Using KAM

Local HPSS methods rely on filtering and diffusion operations and are surprisingly effective for separating percussion instruments from melodic instruments. These methods exploit local characteristics apparent in TF representations of the music mixture (often the magnitude spectrogram). Typically, tonal TF areas are assigned to the harmonic component, and transient TF areas to the percussive component [52, 71, 78, 152]. These simplifications are strongly related to classic signal processing paradigms such as sinusoids & transient & noise modeling [183]. However, they are also problematic, since many drum instruments such as kick drums, tom toms, and cymbals exhibit strong tonal (yet inharmonic) components with relatively slow decay. Since harmonicity is a concept based on relationships between sinusoids positioned within a harmonic series (rather than a narrow frequency neighborhood), local methods are usually not suited to emphasize these characteristics. Thus, it may happen that TF components belonging to the percussion are erroneously treated as tonal. In practice, this often leads to audible leakage of the drums' decay into the harmonic signals. In contrast, the separated drum signals may sound unnatural and exhibit severe audible artifacts. As we will explain in Section 8.2.4, our novel approach can help to recover from these errors to a certain extent.

In Algorithm 2, we detail our variant of KAM-based HPSS [46], been originally proposed in [78] as a generalization of the median-filtering method [71]. The gist of this iterative procedure is to alternate between localized enhancement of percussive and harmonic structures [78] and generalized Wiener filtering [46, 134]. To this end, the estimates $\mathrm{A_h^{(0)}}$ and $\mathrm{A_p^{(0)}}$ are initialized with identical copies of the the mixture spectrogram A. Two filter kernels $\mathcal{I}_\mathrm{p} \in \mathbb{R}_{\geq 0}^{\kappa \times 1}$ and $\mathcal{I}_\mathrm{h} \in \mathbb{R}_{\geq 0}^{1 \times \kappa}$ are used to enhance percussive and harmonic structures, respectively. Kernel $\mathcal{I}_\mathrm{p}$ is Hann-shaped and oriented vertically (column vector). Kernel $\mathcal{I}_\mathrm{h}$ holds the same coefficients in perpendicular orientation (row vector). The kernel width $\kappa \in \mathbb{N}$ determines the smoothing strength (potentially, it could be tuned individually for each kernel). In Algorithm 2, the operator $*$ denotes 2D convolution with appropriate zero padding. This operation yields the percussively and harmonically enhanced estimates $\mathrm{B_p}$ and $\mathrm{B_h}$, respectively. Both $\mathrm{B_p}$ and $\mathrm{B_h}$ are used for Wiener filtering, where the multiplication $\odot$ and division $\oslash$ are performed element-wise.

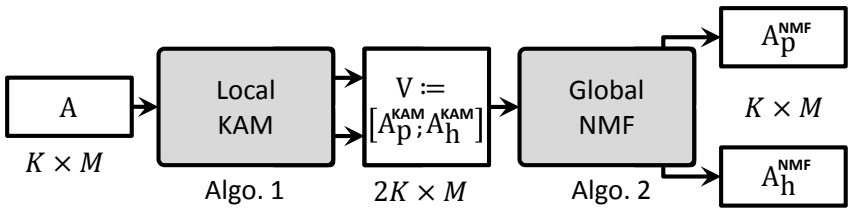

Figure 8.3. Overview of our proposed method.

Algorithm 2: KAM-based HPSS.

Input: $\mathrm{A}_\mathrm{p}^{(0)} := \mathrm{A}$ and $\mathrm{A}_\mathrm{h}^{(0)} := \mathrm{A}$ with $L = L^{\mathrm{KAM}}$
for $\ell = 0, 1, 2, \ldots, L-1$ **do**

$$\left.\begin{aligned} \mathrm{B}_\mathrm{p} &:= \mathrm{A}_\mathrm{p}^{(\ell)} * \mathcal{I}_\mathrm{p} \\ \mathrm{B}_\mathrm{h} &:= \mathrm{A}_\mathrm{h}^{(\ell)} * \mathcal{I}_\mathrm{h} \end{aligned}\right\} \text{2D convolution}$$

$$\left.\begin{aligned} \mathrm{A}_\mathrm{p}^{(\ell+1)} &:= \mathrm{A} \odot \mathrm{B}_\mathrm{p} \oslash (\mathrm{B}_\mathrm{p} + \mathrm{B}_\mathrm{h}) \\ \mathrm{A}_\mathrm{h}^{(\ell+1)} &:= \mathrm{A} \odot \mathrm{B}_\mathrm{h} \oslash (\mathrm{B}_\mathrm{p} + \mathrm{B}_\mathrm{h}) \end{aligned}\right\} \text{Wiener filtering}$$

end
Output: $\mathrm{A}_\mathrm{p}^{\mathrm{KAM}} := \mathrm{A}_\mathrm{p}^{(L)}$ and $\mathrm{A}_\mathrm{h}^{\mathrm{KAM}} := \mathrm{A}_\mathrm{p}^{(L)}$.

8.2.2 Global HPSS Using NMF

Global HPSS approaches factorize the mixture spectrogram into low-rank components and cluster them into percussive and harmonic subsets. As an early example, components uncovered via Independent Subspace Analysis were classified as either percussive or harmonic via spectral and temporal low-level features in [205]. In later works, more suitable factorization techniques such as NMF [24, 25, 102, 119, 155, 156], Non-Negative Tensor Factorization [77], or Non-negative Matrix Factor Deconvolution [39, 125] have been used.

NMF is a popular algorithm for iteratively computing a global, low-rank approximation $\mathrm{V} \approx \mathrm{W} \cdot \mathrm{H}$ [130]. In the context of music source separation, $\mathrm{V} \in \mathbb{R}_{\geq 0}^{K \times M}$ is usually the magnitude spectrogram, the columns of $\mathrm{W} \in \mathbb{R}_{\geq 0}^{K \times R}$ are interpreted as spectral basis functions (also called templates), and the rows of $\mathrm{H} \in \mathbb{R}_{\geq 0}^{R \times M}$ are interpreted as time-varying gains (also called activations).

Since the optimal rank $R \in \mathbb{N}$ is generally unknown and dependent on the content in V, it is commonly set to a sufficiently high number (e. g., to 750 in [156]). In practice, this is problematic since components learned by NMF may represent atomic parts of the individual sources of interest. Automatically clustering them to form musically meaningful parts can be very challenging. For now, let us assume that we already know to which extent each of the R components contributes to the percussive part. Formally, we express this by introducing a percussive weight vector $\mathrm{p} \in \mathbb{R}_{\geq 0}^{1 \times R}$, with entries $0 \leq \mathrm{p}(r) \leq 1$, $r \in [1 : R]$. The values of $\mathrm{p}(r)$ define a continuum between the percussive ($\mathrm{p}(r) = 1$) and harmonic ($\mathrm{p}(r) = 0$) extremes. With this pre-requisite, it is

straightforward to construct the percussive and harmonic estimates as:

$$\begin{aligned} \mathrm{V}_{\mathrm{p}}^{\mathrm{NMF}} &:= \mathrm{W} \cdot (\mathrm{P} \odot \mathrm{H}) \,, \\ \mathrm{V}_{\mathrm{h}}^{\mathrm{NMF}} &:= \mathrm{W} \cdot ((1 - \mathrm{P}) \odot \mathrm{H}) \,, \end{aligned} \tag{8.1}$$

where the percussive weight matrix $\mathrm{P} \in \mathbb{R}_{\geq 0}^{R \times M}$ just replicates the elements of p over all columns. In Figure 8.2b and Figure 8.2c, we show activations and templates for our running example, the shades of gray encoding $\mathrm{p}(r)$ as given in Figure 8.2d. We will explain how to determine $\mathrm{p}(r)$ in Section 8.2.4. Note that that the first four drum-related components exhibit decaying impulses at the time instances corresponding to drum hits. In contrast, the remaining sax-related components exhibit plateau-like activations when notes are being played.

8.2.3 Soft Activation Constraints

Many authors proposed to apply constraints to NMF in order to guide the iterative factorization process towards a meaningful solution. Usually, these constraints are applied in between the regular NMF update rules by manipulating both W and H. For example, gamma priors were used in [77] to encourage temporal continuity in the activations of harmonic components that are assumed to vary slowly in time. Similarly, constraints promoting smoothness of the activations and sparseness of the templates were used in [24]. A method for emphasizing harmonic structures in some templates was proposed in [155], while spectral and temporal continuity constraints were introduced via a weighted sum scheme in [156].

In this chapter, we propose two non-linear operations which are applied exclusively to the individual rows of H prior to the NMF updates. Analogous to the HPSS method by Fitzerald [71], harmonically enhanced activations H_h are computed by median filtering along the horizontal axis as:

$$\mathrm{H}_h(r, m) := \mathrm{median}\left(\mathrm{H}(r, m - \tau), \ldots, \mathrm{H}(r, m + \tau)\right), \tag{8.2}$$

for $\tau \in \mathbb{N}$ with $2\tau + 1$ being the length of the median filter. In the following, we will use the operator $\mathcal{H}(\cdot)$ to denote this operation. In Figure 8.2b, $\mathcal{H}(\cdot)$ would emphasize the plateau-like shapes in the activations of the alto sax while suppressing impulse-like patterns.

In contrast, percussively enhanced activations H_{p} are derived by applying a non-linear exponential moving average filter (NEMA). Setting the first element $\mathrm{H}_{\mathrm{p}}(r, 0) := \mathrm{H}(r, 0)$, the NEMA operation applied to the r^{th} row can be described as follows:

$$\mathrm{H}_{\mathrm{p}}(r, m) := \max \begin{cases} \mathrm{H}(r, m) \\ \lambda \cdot \mathrm{H}_{\mathrm{p}}(r, m - 1) + (1 - \lambda) \cdot \mathrm{H}(r, m) \end{cases} \tag{8.3}$$

for $m \in [1 : M-1]$. The weight $\lambda \in \mathbb{R}$ with $0 < \lambda < 1$ controls the decay of this recursive filter, which we denote by $\mathcal{P}(\cdot)$ in the following. As illustrated in Figure 8.2b, $\mathcal{P}(\cdot)$ promotes the development of sharp peaks followed by a moderate decay. Contrary to the common belief that impulse-like NMF activations are suited to model drum sound events, we showed in [39] that decaying impulses are more appropriate for music source separation.

Algorithm 3: NMF-based HPSS.

Input:
Concatenate KAM results $\mathrm{V} := \left[\mathrm{A}_{\mathrm{p}}^{\mathrm{KAM}}; \mathrm{A}_{\mathrm{h}}^{\mathrm{KAM}}\right]$
Initialize all-ones matrix $\mathrm{J} \in \mathbb{R}_{\geq 0}^{2K \times M}$
Initialize $\mathrm{H}^{(0)}$ and $\mathrm{W}^{(0)}$ with non-negative random values
Set $L = L^{\mathrm{NMF}}$
for $\ell = 0, 1, 2, \ldots, L-1$ **do**
  Define P from $\mathrm{W}^{(\ell)}$ via (8.4)
  $\mathrm{B} := \mathrm{P} \odot \mathcal{P}\left(\mathrm{H}^{(\ell)}\right) + (1 - \mathrm{P}) \odot \mathcal{H}\left(\mathrm{H}^{(\ell)}\right)$
  $\left.\begin{aligned} \mathrm{H}^{(\ell+1)} &:= \mathrm{B} \odot \frac{\mathrm{W}^{(\ell)\top} \frac{\mathrm{V}}{\mathrm{W}^{(\ell)}\mathrm{B}}}{\mathrm{W}^{(\ell)\top}\mathrm{J}} \\ \mathrm{W}^{(\ell+1)} &:= \mathrm{W}^{(\ell)} \odot \frac{\frac{\mathrm{V}}{\mathrm{W}^{(\ell)}\mathrm{B}}\mathrm{B}^{\top}}{\mathrm{J}\mathrm{B}^{\top}} \end{aligned}\right\}$ NMF updates
end
Set $\mathrm{W} := \mathrm{W}^{(L)}$ and $\mathrm{H} := \mathrm{H}^{(L)}$
Binarize p according to threshold p_{thr}
Compute $\mathrm{V}_{\mathrm{p}}^{\mathrm{NMF}}$ and $\mathrm{V}_{\mathrm{h}}^{\mathrm{NMF}}$ via (8.1)
Output:
$\mathrm{A}_{\mathrm{p}}^{\mathrm{NMF}} := \mathrm{A} \odot \left(\Lambda \cdot \mathrm{V}_{\mathrm{p}}^{\mathrm{NMF}} \oslash \left(\mathrm{V}_{\mathrm{p}}^{\mathrm{NMF}} + \mathrm{V}_{\mathrm{h}}^{\mathrm{NMF}}\right)\right)$
$\mathrm{A}_{\mathrm{h}}^{\mathrm{NMF}} := \mathrm{A} \odot \left(\Lambda \cdot \mathrm{V}_{\mathrm{h}}^{\mathrm{NMF}} \oslash \left(\mathrm{V}_{\mathrm{p}}^{\mathrm{NMF}} + \mathrm{V}_{\mathrm{h}}^{\mathrm{NMF}}\right)\right)$

8.2.4 Unifying KAM and NMF

In Algorithm 3, we detail the unification of KAM-based HPSS with our soft-constrained NMF. In contrast to prior approaches, we start by stacking the KAM-based estimates of percussive and harmonic parts into a concatenated matrix $\mathrm{V} \in \mathbb{R}_{\geq 0}^{2K \times M}$, with $\mathrm{V} := \left[\mathrm{A}_{\mathrm{p}}^{\mathrm{KAM}}; \mathrm{A}_{\mathrm{h}}^{\mathrm{KAM}}\right]$. This matrix is then used as the target for NMF decomposition. Consequently, our NMF bases can be imagined as stacked templates of dimension $\mathrm{W} \in \mathbb{R}_{\geq 0}^{2K \times R}$. This core idea of our novel approach serves the following two purposes.

First, it enables the redistribution of TF magnitude that had been assigned to the wrong part by KAM. The rationale is that in our framework, a single NMF template can be interpreted as a coupling between two templates. The first template (corresponding to the lower K frequency bins) can only "see" the percussive estimate, while the second template (corresponding to the upper K frequency bins) can only model spectral patterns contained in the harmonic estimate.

Since the coupled templates share one activation, they both can collect spectral contributions according to the activation. In Figure 8.2c, this effect is illustrated by the fact that the drum templates have been assigned considerable contributions from the harmonic part while the sax templates have been assigned transient spectra (note the smeared harmonic structure) from the percussive estimate.
Second, our approach enables straightforward estimation of the percussive weight p from the relationship between the lower half (percussive) and the upper half (harmonic) of the NMF templates. Formally, $\mathrm{p}\,(r)$ is given as:

$$\mathrm{p}\,(r) := \frac{\sum_{k=0}^{K-1} \mathrm{W}(k,r)}{\sum_{k=0}^{2K-1} \mathrm{W}(k,r)}, \tag{8.4}$$

for each of the r^{th} NMF components. In Algorithm 3, replication of p over all columns yields the percussive weight matrix P. This matrix is then used to construct the weighted superposition B of the latest percussively and harmonically enhanced activations. The NMF updates use B instead of the regular $\mathrm{H}^{(\ell)}$. Since the percussive weight vector p depends on the templates W, it implicitly classifies the NMF components. In contrast to [156], this classification is not pre-defined — it is soft, and it adapts to the components as they evolve during the NMF iterations.
Finally, after the NMF iterations have reached the limit L^{NMF}, we achieve the final classification of the components by binarizing p according to a pre-defined threshold $p_{\mathrm{thr}} \in [0, 1]$. This step is necessary to achieve a good separation between the refined NMF components. It remains to be seen whether more elaborate classification schemes are beneficial.
A final Wiener filtering step then delivers the desired percussive part $\mathrm{A}_{\mathrm{p}}^{\mathrm{NMF}}$ and harmonic part $\mathrm{A}_{\mathrm{h}}^{\mathrm{NMF}}$. To make this work, we need to revert the earlier spectrogram stacking by multiplication with an aggregation matrix $\Lambda \in \mathbb{R}_{\geq 0}^{K \times 2K}$, constructed as $\Lambda := [\mathrm{I}, \mathrm{I}]$, with $\mathrm{I} \in \mathbb{R}_{\geq 0}^{K \times K}$ being the identity matrix.

8.3 Evaluation

In this section, we present some HPSS experiments to compare our proposed approach to other state-of-the-art methods. To this end, we composed 74 music mixtures using two datasets, the ENST-Drums[35] corpus and the QUASI[36] corpus, both containing multi-track music recordings with ground-truth drum parts. We use an STFT blocksize of 2048 samples (approx. 46 ms) and a hopsize of 512 samples (75 % overlap). Table 8.1 summarizes the comparison algorithms and their parameters. Note that the outcomes of KAM serve as initial estimates of our proposed method.

[35] `http://perso.telecom-paristech.fr/~grichard/ENST-drums/`, last accessed June 14, 2018
[36] `http://www.tsi.telecom-paristech.fr/aao/en/2012/03/12/quasi/`, last accessed June 14, 2018

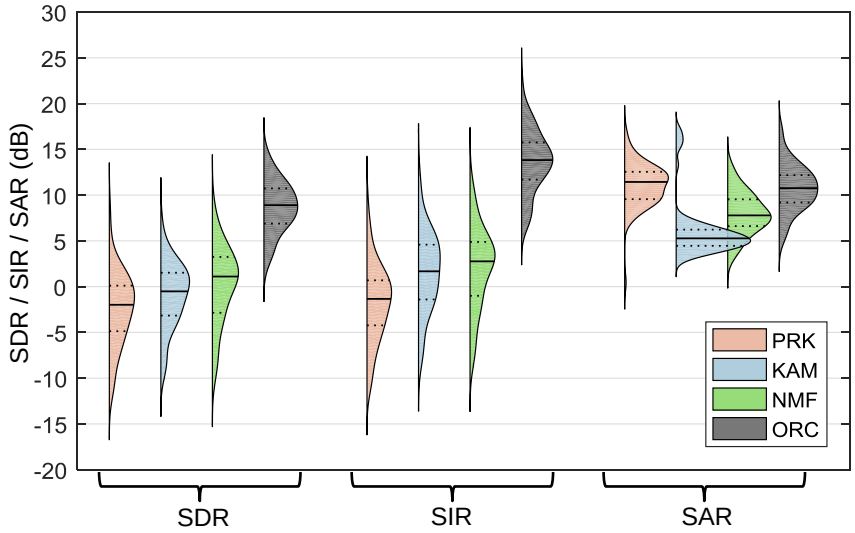

Figure 8.4. BSS Eval comparison of the HPSS methods listed in Table 8.1. The test set comprises the corpora ENST-Drums and QUASI. The thick solid line in each distribution shows the median value over all items, and the dashed lines delimit the corresponding interquartile range.

Algorithm	**Details**
PRK: Constrained NMF [156]	$L^{\mathrm{NMF}} = 100$, $R = 750$, continuity parameters from [156]
KAM: KAM-based HPSS [46]	$L^{\mathrm{KAM}} = 30$, $\kappa = 9$, Hann-shaped kernel
NMF: Proposed Method	$L^{\mathrm{NMF}} = 60$, $R = 30$, $p_{\mathrm{thr}} = 0.25$, Median: $\tau = 4$, NEMA: $\lambda = 0.75$
ORC: Oracle [134]	Wiener Filtering using true source spectrograms

Table 8.1. Configuration of the test cases in our comparative performance evaluation.

All algorithms deliver estimates for the percussive and harmonic magnitude spectrograms. We reconstruct the corresponding time-domain signals via inverse STFT using the mixture phase. We measure the separation quality using the median BSS Eval metrics SDR, SIR, and SAR [209]. As can be seen in Figure 8.4, our proposed method surpasses the comparison methods KAM [46] and PRK [156] in the majority of metrics. However, it still falls short of the oracle Wiener filtering (ORC) presented in Figure 8.4c.

8.4 Discussion and Outlook

Since BSS Eval provides objective measures, we recommend listening to the audio examples on our accompanying website[37] to get a better impression of the capacities and limitations of our proposed method. The examples are taken both from real-world music recordings (including the excerpt in Figure 8.2), as well as from our test set. With KAM, one can hear that the decay and reverb of the drum part is often assigned to the harmonic component. The method by Park et al. [156] better preserves the drum characteristics but often has considerable leakage of the melodic instruments' attack into the drums. With our method, we are able to improve the quality of the drum signal considerably, while still achieving moderate separation. On the downside, the audio examples reveal that our method has difficulties to model quickly varying melodic signals, such as singing voice. Moreover, it is susceptible to distorted guitars, which produce spectra that look more broadband and noise-like than pure melodic tones.
Future work will be concerned with more thorough, data-driven parameter optimization. We plan to investigate using additional side-information (e. g., drum-specific templates), which can be easily integrated to guide the NMF updates [39, 67]. Also, it might be beneficial to tune the decay parameter λ depending on the underlying instrument (e.g., longer decay for cymbals and kick drums).

[37] `https://www.audiolabs-erlangen.de/resources/MIR/2018-ICASSP-HPSS_KAM_NMF/`, last accessed June 14, 2018

Part III

Applications to Jazz Research

Chapter 9

Swing Ratio Estimation

The work in this chapter is mainly based on our contribution in [44]

In this chapter, we propose a new method suitable for the automatic analysis of microtiming played by drummers in jazz recordings. Specifically, we aim to estimate the drummers' swing ratio in music excerpts contained in the Weimar Jazz Database. A first approach is based on automatic detection of ride cymbal (RC) onsets and evaluation of relative time intervals between them. However, small errors in the onset detection propagate considerably into the swing ratio estimates. As our main technical contribution, we propose to use the log-lag autocorrelation function (LLACF) as a mid-level representation for estimating swing ratios, circumventing the error-prone RC transcription step. In our experiments, the LLACF-based swing ratio estimates prove to be more reliable than the ones based on RC onset detection. Therefore, the LLACF seems to be the method of choice to process large amounts of jazz recordings. Finally, we indicate some implications of our method for microtiming studies in jazz research.

9.1 Introduction

Jazz drummers usually keep time by using the ride cymbal (RC) and hi-hat (HH), especially in styles with so-called "swing feel"[16]. They commonly emphasize the "onbeat," i.e., the metric-harmonically unaccented beat, on the HH while playing typical patterns on the RC. According to [163, p. 248], this supports the "light" character of jazz rhythm. Instead of playing the beat in a steady manner, variations and additional "offbeat" strokes are usually added on the RC as well as on other drum parts. These variations differ from drummer to drummer and from performance to performance [16, pp. 617-629].
The most common time-keeping pattern played on the RC is shown in Figure 9.1. In addition to

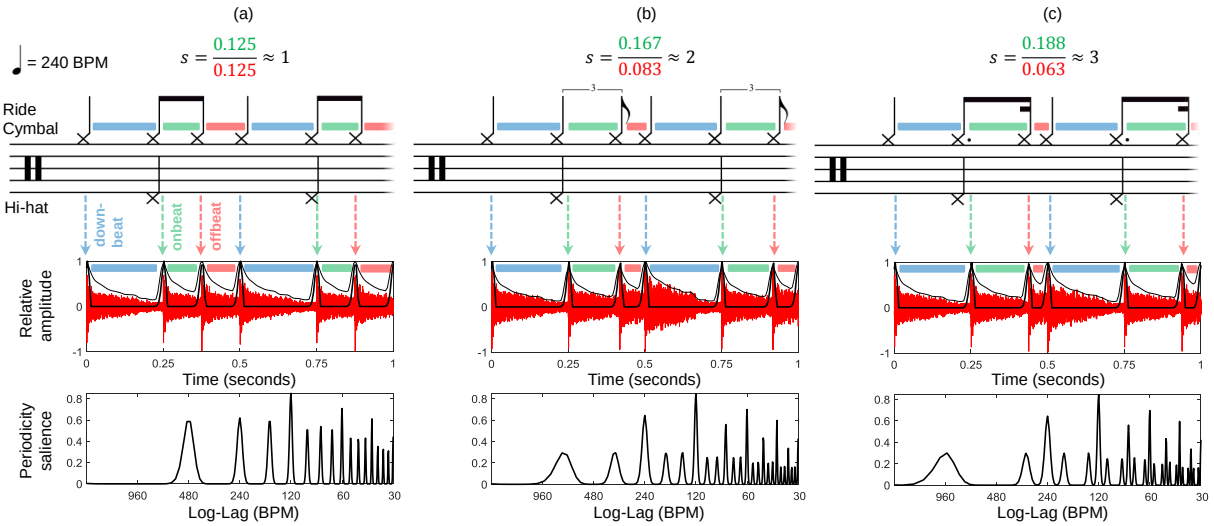

Figure 9.1. Illustration of prototypical RC patterns as drum notation (top), time-domain signal (mid), and LLACF (bottom). **(a)** Swing ratio of $s = 1$ corresponding to straight eighth-notes. **(b)** Swing ratio of $s = 2$ corresponding to the idealized "tied-triplet feel". **(c)** Swing ratio $s = 3$, where the inter-onset-interval of the onbeat equals a dotted eighth-note.

conventional drum notation in the top row, we show a corresponding time-domain signal (in red) with overlaid amplitude envelope (thin black curve) and the so-called novelty curve (bold black curve). We color-code the relevant beats and subdivisions thereof as follows. The sequence starts with the so-called "downbeat" quarter note (light blue), followed by the onbeat eighth-note (light green), and the offbeat eighth-note (light red) before starting over again with the downbeat. We will refer to this prototype sequence of onsets as RC pattern.

The so-called swing ratio expresses the beat subdivision and relates to the phrasing of the eighth-notes in the RC pattern. Swinging eighth-notes are typically played in different ratios, ranging continuously from straight eighths $(1:1)$, over triplet eighths $(2:1)$, to dotted eighths $(3:1)$, or more extreme ratios. The swing ratio is reported to be tempo dependent [23, 81, 113], cf. Section 9.2.1. In Figure 9.1, the color-coded tone bars show how the onbeat inter-onset-interval (IOI) grows with increasing swing factor, while the complementing offbeat IOI shrinks. In Figure 9.1a, onbeat and offbeat have equal IOIs, corresponding to straight eighths as given in the drum notation. In Figure 9.1b, the RC pattern is notated as tied-triplets. In Figure 9.1c, the onbeat IOI equals a dotted eighth. Consequently, the offbeat IOI equals that of a sixteenth-note as shown in the drum notation.

There are several case studies concerning the swing ratio in jazz (cf. [163, pp. 262-273], and Section 9.2.1). While most of the studies examine swing ratios of soloists, it is widely acknowledged that the swing ratio of the RC pattern crucially contributes to the "swinging" character of the music. Most of the studies are based on manual transcription of onsets, often by visual inspection of the amplitude envelope of jazz excerpts. Few studies specifically examine the RC

pattern [113] and its interaction with the soloist's timing [81]. This inspired us to develop and to evaluate methods for automated swing ratio estimation from RC patterns in jazz recordings. For sure, an automated generation of large amounts of reliable swing ratio data is essential for meaningful and more differentiated research on microtiming in jazz. Besides onset-based swing ratio estimation, our main approach is a log-lag variant of a local autocorrelation function (ACF) applied to onset-related novelty functions (see Section 9.3.3). We refer to this representation as log-lag ACF (LLACF) and show its applicability to swing ratio estimation in Section 9.3.4.

9.2 Related Work

A number of papers are concerned with systematic studies on swing ratio in jazz music. Since most of the studies use comparably small datasets and manual annotation, we think that swing ratio estimation is a suitable task to apply automatic methods from Music Information Retrieval (MIR) research in order to enable analysis of larger music datasets.

9.2.1 Jazz Microtiming Analysis

An early attempt to analyze swing ratios in jazz recordings is described in [118]. The author relies on visual inspection of spectrograms but does not report quantitative results. In [170], the swing ratios in the analyzed jazz recordings are reported to range from 1.48 to 1.82. Rose [175] reports an average swing ratio of 2.38 measured from amplitude envelopes. In [61], an average swing ratio of 1.75 is measured using a MIDI wind controller played by saxophonists. In [157], the analysis focuses on the RC and swing factors between 1.0 and 3.3 are reported without detailing the measurement method. In [32], an average swing ratio of 1.6 is measured using amplitude envelopes. Friberg and Sundström [81] annotated RC onsets in spectrograms of jazz excerpts. They report trends indicating a high negative correlation between the tempo and the swing ratio which seems to be valid across different drummers. In [22], an average swing ratio of 2.45 is measured in the performances of pianists playing a MIDI piano. In [10], comparably low swing ratios in the range between 0.9 to 1.7 are measured from amplitude envelopes. Honing and de Haas [113] conducted experiments with professional jazz drummers performing on a MIDI drum kit. Besides further evidence for the tempo dependency of swing ratios, the results show that jazz drummers have enormous control over their timing.

9.2.2 Rhythmic Mid-Level Features

Motivated by the need to design specialized mid-level features for music similarity estimation, several authors proposed conceptually similar, tempo-independent representations of rhythmic

patterns. The basic observation is, that rhythmic patterns that are perceived as similar by human listeners may not be judged as similar by automatic methods. One of the main reasons is that the patterns are typically played in different tempi, which makes them unsuited for direct comparison. Therefore, Peeters [161] used tempo normalized spectral rhythm patterns to automatically classify ballroom dance styles. Holzapfel and Stylianou [111, 112] proposed to apply the scale transform to periodicity spectra to enable the use of conventional distance measures between rhythmic patterns despite tempo differences. Around the same time, the LLACF was proposed in [100] as well as the tempo-insensitive representation used for classification of ballroom dances in [114]. The LLACF was reported to be favorable over the scale transform for classification of Latin American rhythm patterns in [214]. The tempogram as described in [99] is based on similar ideas and additionally features a cyclic post-processing to remedy the problem of octave ambiguity. Marchand and Peeters [139] revisited the scale transform and applied it to modulation spectra as tempo-independent feature, again for classification of ballroom dances. Eppler et al. [64] used peak ratios in the LLACF as features for detecting the swing feel but did not explicitly try to estimate swing ratios.

9.3 Method

In this section, we describe our approaches to automatic swing ratio estimation from excerpts of jazz recordings with swing feel. The first variant relies on peak-picking in an onset-related novelty curve (Section 9.3.1). The second approach relies on computation of the LLACF from the novelty curve (Section 9.3.3) and comparison to prototype LLACFs. As will be explained in Section 9.4.1, we have a rough tempo estimate $t_{\mathrm{e}} \in \mathbb{R}_{>0}$ available for each jazz excerpt. Let $\delta_a \in \mathbb{R}_{>0}$ be the onbeat IOI and $\delta_b \in \mathbb{R}_{>0}$ the offbeat IOI in an RC pattern as shown in Figure 9.1. They relate to the tempo by $t_{\mathrm{e}} \approx (\delta_a + \delta_b)^{-1} \approx \delta_{a+b}^{-1}$, with the beat (quarter note) IOI $\delta_{a+b} \in \mathbb{R}_{>0}$. The targeted swing ratio is given by:

$$s = \frac{\delta_a}{\delta_b} \tag{9.1}$$

Consequently, $\delta_a = \delta_{a+b} \cdot s \cdot (1+s)^{-1}$ yields the onbeat IOI and $\delta_b = \delta_{a+b} \cdot (1+s)^{-1}$ yields the offbeat IOI.

9.3.1 Ride Cymbal Onset Detection

With regard to (9.1), we aim to measure δ_a and δ_b from the jazz excerpts under analysis. One possibility is to search for RC onsets and use the time differences between consecutive onsets as IOI estimates. To this end, we compute a time-frequency (TF) representation of an excerpt using the short-time Fourier transform (STFT) with blocksize $N \in \mathbb{N}$ and hopsize $H \in \mathbb{N}$. Let $\mathcal{X}(m,k)$ with $m \in [1:M]$, $k \in [0:K]$ be a complex-valued STFT coefficient at the m^{th}

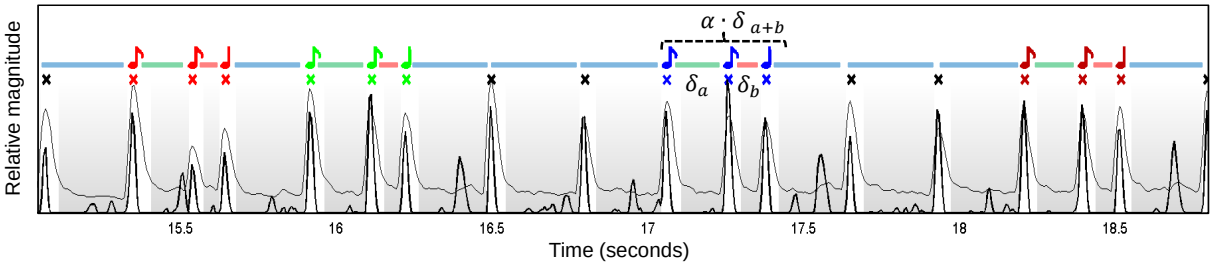

Figure 9.2. A four seconds excerpt from the 1979 recording of "Anthropology", performed by Art Pepper playing solo clarinet, with Charlie Haden on bass and Billy Higgins on drums. The bold black curve depicts the novelty function Δ, the thin black curve shows the RC related threshold Λ. Automatically detected RC onsets are marked by the bold black crosses, colored crosses represent the four onset triples accepted for swing ratio estimation. The IOIs are color-coded in the same way as in Figure 9.1.

time frame and k^{th} spectral bin. Here, the interval $[1:M]$ represents the time axis and K corresponds to the Nyquist frequency. Following the approaches in [97, 99], we compute a novelty curve $\Delta : [1:M] \to \mathbb{R}$ as follows. First, we derive the logarithmically compressed magnitude spectrogram $\mathcal{Y}(m,k) := \log\left(1 + \gamma \cdot |\mathcal{X}(m,k)|\right)$ for a suitable constant $\gamma \geq 1$. Then, the novelty function is given as

$$\Delta(m) := \sum_{k=0}^{K} |\mathcal{Y}(m+1,k) - \mathcal{Y}(m,k)|_{\geq 0}\,, \tag{9.2}$$

where $|\cdot|_{\geq 0}$ denotes half-wave rectification. The resulting Δ exhibits salient peaks at frames corresponding to tone onsets. Inevitably, spurious peaks may occur in Δ that could be mistaken for RC onsets. Thus, we derive an RC related threshold function as

$$\Lambda(m) := \sum_{k=k_0}^{K} |\mathcal{X}(m,k)|\,, \tag{9.3}$$

where the bin k_0 corresponds to the lower cutoff frequency. Figure 9.2 shows an example of Δ as bold black curve and the corresponding Λ as thin black curve. For the sake of visibility, both curves are normalized to unit maximum in the plot. We take the average value of Λ as threshold criterion and only accept peaks from Δ in frames where Λ exceeds this value (indicated by the white background). The $S = 18$ local maxima accepted as RC onsets are marked by bold crosses. Multiplication of the corresponding frame indices with the temporal resolution H/f_s (f_s is the given sampling rate) yields a set of strictly monotonically increasing onset times $B = \{b_1, b_2, \ldots, b_S\}$ for onset-based swing ratio estimation.

9.3.2 Onset-Based Swing Ratio Estimation

Once we obtained a sequence B of RC onsets, we estimate s in a tempo-informed manner. Assuming a roughly constant tempo t_e throughout the excerpt, the time interval $\delta_{a+b} = t_\mathrm{e}^{-1}$ between two consecutive beats should be close to $\delta_a + \delta_b$. To account for small deviations from the ideal beat period δ_{a+b}, we introduce a tolerance $\alpha \geq 1$. Now, we go through every previously detected RC onset and test the hypothesis that it could be the first in a series of three consecutive onsets (onbeat, offbeat, downbeat). We denote this sub-sequence as $B_s = \{b_s, b_{s+1}, b_{s+2}\}, B_s \subset B$ and refer to it as onset triple. From all possible triples $B_s, s \in [1 : S-2]$ we accept the ones that fulfill the criterion

$$(b_{s+2} - b_s) < \alpha \cdot \delta_{a+b} \tag{9.4}$$

as instances of triples embedded in an RC pattern. The swing ratio is estimated from a valid onset triple by setting $\delta_a = b_{s+1} - b_s$ and $\delta_b = b_{s+2} - b_{s+1}$ in (9.1). In Figure 9.2, we illustrate this procedure. All RC onset candidates are marked by black crosses but only the triples that fulfill the constraint in (9.4) are marked with different colors. Above the third triple (blue note symbols) we depict the extent of the search range $\alpha \cdot \delta_{a+b}$ that covers both δ_a and δ_b. As indicated in the plot, we try to find multiple occurrences of the RC pattern triples per excerpt, so we can obtain a more robust estimate for the swing ratio by averaging over the individual s-values computed for each triple. For that reason, we also accept variations of the RC pattern where the offbeat impulse occurs in succession to the downbeat instead of the onbeat. As will be explained in Section 9.4.4, there are situations where estimation of s from RC onsets may deliver erroneous results. To obtain more robust estimates, we introduce LLACF-based swing ratio estimation in the next two sections.

9.3.3 LLACF Mid-Level Representation

We propose to employ the LLACF as a tempo-normalized mid-level representation capturing the swing ratio that is implicitly encoded in the peaks of Δ. Using the LLACF, we can circumvent the selection of onset candidates and instead transform the complete Δ into a phase-invariant, tempo-normalized representation. Swing ratio estimation then boils down to matching this representation to LLACFs with known swing ratios (see Section 9.3.4). To this end, we first compute a normalized ACF from the novelty function Δ as:

$$R_{\Delta\Delta}(\ell) = \frac{\sum_{m=1}^{M-\ell} \Delta(m)\Delta(m-\ell)}{\sum_{m=1}^{M} \Delta(m)^2}, \tag{9.5}$$

where we only consider the positive lags $\ell \in [0 : M-1]$. Note that $R_{\Delta\Delta}(\ell) = R_{\Delta\Delta}(-\ell)$ due to symmetry. Moreover, $R_{\Delta\Delta}(0) = 1$ and $R_{\Delta\Delta}(\ell) < 1$ for $\ell \in [1 : M-1]$. Each lag can be expressed as tempo value by the relation $t = \frac{f_s \cdot 60}{H \cdot \ell}$. We now define a logarithmically spaced tempo

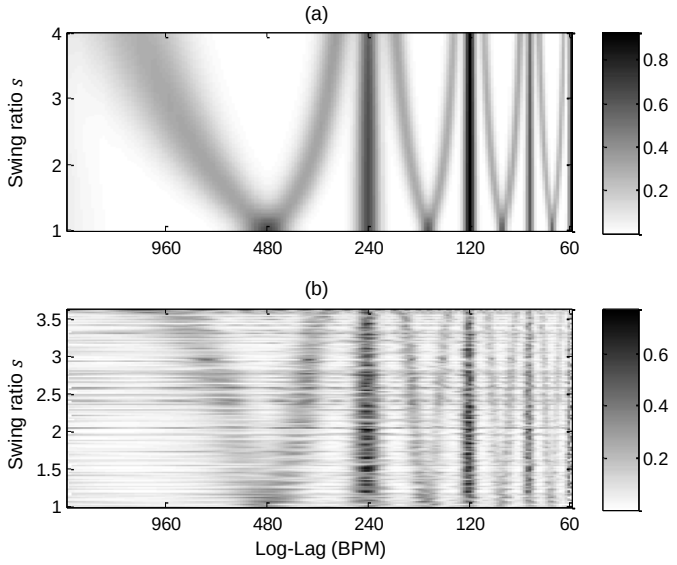

Figure 9.3. Evolution of the LLACF computed from RC patterns with increasing swing ratio. **(a)** LLACFs derived from novelty functions of idealized prototype RC patterns at a reference tempo t_r of 240 BPM. **(b)** LLACFs extracted from our test corpus that have been warped to match t_r.

(log-tempo) axis, that has equal distance q between tempo octaves and has the reference tempo t_r at a defined position. After correction for the ratio between the excerpt's tempo estimate t_e and the reference tempo t_r, we use linear interpolation to warp $R_{\Delta\Delta}$ onto this axis, yielding our tempo-normalized LLACF x. Despite using a log-tempo axis, we stick to the term log-lag ACF since the inverse relation $\ell = \frac{f_s \cdot 60}{H \cdot t}$ retains the logarithmic spacing, just in opposite direction.
In the bottom row of Figure 9.1, we show the LLACFs corresponding to the prototypical RC patterns. Variation of s gives an intuition how the salience of different periodicities in the RC pattern is represented by the LLACF. Since t_r is constant, all three LLACFs have clear peaks at the beat periodicity (240 BPM) and its integer subdivisions. For $s = 1$ in Figure 9.1a, there is a strong peak at 480 BPM (corresponding to the straight eighth-notes). With increasing swing ratio, this peak diverges into two lobes that move to other periodicities. In Figure 9.1c, the first peak resides at 960 BPM (offbeat equals a sixteenth-note) and the second peak is at 320 BPM (onbeat equals a dotted eight note).

9.3.4 LLACF-Based Swing Ratio Estimation

In order to estimate a swing ratio from the shape of x, we construct a set $\mathrm{Y}_s, s \in \mathbb{R}$ with $1 \leq s \leq 4$ of prototype LLACFs. They are extracted from novelty functions of idealized RC patterns with fixed reference tempo t_r and varying swing ratio s (cf. the time-domain plots in Figure 9.1). In Figure 9.3a, we show the complete set of prototype LLACFs with the log-tempo axis in BPM and the swing ratio increasing from bottom to top. Darker shade of

gray corresponds to higher periodicity salience. One can clearly see how the offbeat-related peaks change their periodicity with the swing ratio while the peaks related to the beat (and subdivisions thereof) reside at the same periodicity.
Now, our approach to swing ratio estimation is to compare the extracted x to each of these prototype LLACFs and to select the swing ratio corresponding to the best match. For the comparison, we employ Pearson's correlation coefficient. We have to take into account that the tempo estimate t_e used for warping the LLACF to the reference log-tempo axis underlying Y_s may be slightly inaccurate. As a consequence, the resulting x might exhibit a constant offset with respect to the prototype Y_s. Thus, we shift the x against the log-tempo axis of each Y_s in a restricted interval $[-q \cdot \log_2(\alpha) : +q \cdot \log_2(\alpha)]$ to find the best alignment. Finally, the s corresponding the maximum correlation coefficient over all entries in Y_s is selected.

9.4 Evaluation

In this section, we describe the setup, metrics, and results of the experiments we conducted in order to compare manual, onset-based, and LLACF-based swing ratio estimation. In addition, some trends visible in the data are discussed.

9.4.1 The Weimar Jazz Database

The Weimar Jazz Database (WJD[38]) consists of 299 (as in July 2015) transcriptions of instrumental solos in jazz recordings performed by a wide range of renowned jazz musicians. The solos have been manually annotated by musicology and jazz students at Liszt School of Music Weimar as part of the Jazzomat Research Project.[39] Several music properties are annotated, most notably the pitch, onset and offset of all tones played by the soloists, as well as a manually tapped beat grid, chords, form parts, phrase boundaries, and articulation. For our work, we only use the beat grid. From the complete WJD, we automatically selected a subset of 921 excerpts that had been labeled with swing feel. Because we will compare the swing ratios of drummers and soloists in our future work, the excerpts had to contain at least 5 consecutive eighth-notes played by the soloists. The total playtime of the selected excerpts amounts to roughly 50 minutes (out of 8 hours), their average duration is 3.3 seconds.

[38] `http://jazzomat.hfm-weimar.de/dbformat/dboverview.html`, last accessed June 14, 2018
[39] `http://jazzomat.hfm-weimar.de/`, last accessed June 14, 2018

9.4.2 Evaluation Setting

A subset of 42 excerpts have been manually annotated for RC onsets in order to create a ground truth for swing ratio estimation. The reference onsets were transcribed by two experienced student assistants of the Jazzomat Research Project using the software Sonic Visualiser [26]. The ground truth subset was split in two, approximately equal parts and each part was given to one of the annotators. In total, 834 RC onsets were manually annotated. In our evaluation (cf. Sections 9.4.3, 9.4.4, and 9.4.5), we used the wellknown metrics recall, precision and F-measure for reporting quantitative results. In order to count an onset candidate as true positive, we allowed a maximum deviation of ±30 ms to the ground truth onset time. Furthermore, we used Pearson's correlation coefficient as a means to quantify the agreement between reference swing ratios and automatically estimated swing ratios. We fixed the following extraction parameters for the automatic estimation of swing ratios: The STFT blocksize N was appr. 46 ms and the temporal resolution H/f_s was appr. 5.8 ms. The compression-constant γ was 1000, the lower cutoff k_0 was set to equal appr. 12.9 kHz, the reference tempo t_r was 240 BPM, the LLACF octave-resolution q was 36. The tolerance α for tempo deviations was 1.2.

9.4.3 Cross-Validation

At first, we are interested in the agreement between our human annotators, since we suspect that there may be ambiguous cases where it is not clear where an RC onset is exactly located in time or if there is an onset at all. Thus, we selected a small subset of 11 excerpts for which the annotators created a cross-validation transcription. Running these against the larger set, we receive an F-measure of appr. 0.96. The average absolute time difference between matched onsets in the reference and the cross-validation set amounts to 7.8 ms.

9.4.4 Onset-Based Evaluation

Next, we used the previously validated ground truth annotations as reference to assess the performance of our automated RC onset detection described in Section 9.3.2. In this scenario, we received an F-measure of appr. 0.93 and an average onset deviation of 2.5 ms. Since these results seem surprisingly good, we wanted to quantify how much potential onset detection errors would propagate into the swing ratio estimation. Using the procedure described in Section 9.3.2, we determined ground truth swing ratios for all manually annotated excerpts. When we compared these to the swing ratios estimated from automatically detected RC onsets, we yielded a correlation coefficient of appr. 0.66 (see Figure 9.4). With regard to the comparably high F-measure obtained for the onset detection, this unsatisfactory result may seem surprising at first, but can be explained using the example in Figure 9.2. There, we see that only 12 out

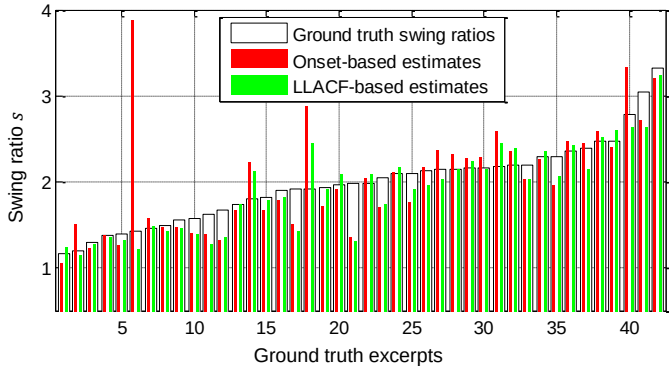

Figure 9.4. Comparison of the swing ratios estimated from ground truth RC onsets, automatically detected RC onsets and LLACF analysis.

of 18 RC onsets are considered for swing ratio estimation. Intuitively, small deviations in the detected onset times can lead to under- or overestimation of the swing ratio, especially for fast tempi, where subtle timing differences may get lost due to the coarse sampling of the analysis frames. Even worse errors may be caused by spurious onsets that fulfill the threshold criterion but are actually not RC patterns. This is the case for the sixth excerpt in Figure 9.4, where some sort of RC swell is mistaken for an onset triple, leading to a overestimation of s.

9.4.5 LLACF-Based Evaluation

Since we found the correlation between ground truth swing ratios and onset-based swing ratios to be unsatisfactory, we repeated the comparison with respect to swing ratios estimated from the LLACF as described in Section 9.3.3. This time, we received a correlation coefficient of appr. 0.9. In Figure 9.4, one can see that both methods behave similar but the onset-based swing ratios exhibit some pronounced outliers. Moreover, Figure 9.3 shows that the prototypical LLACFs in Y_s correspond quite well to the LLACFs extracted from our test corpus. Both plots depict the LLACFs ordered by the corresponding swing ratio. The typical structure of periodicity peaks is clearly visible, although the LLACFs extracted from the jazz excerpts are much more noisy than the idealized LLACFs. This leads us to the conclusion that the LLACF-based swing ratio estimation is a reliable method that should be preferred over the onset-based swing ratio estimation.

9.4.6 Comparison to Friberg and Sundström

In Section 9.1, we already indicated our aim to re-examine the findings of Friberg and Sundström [81] on a larger scale. As can be seen in Figure 9.5a, our automatically estimated swing ratios show similar trends as the manually annotated data used in the original paper. However,

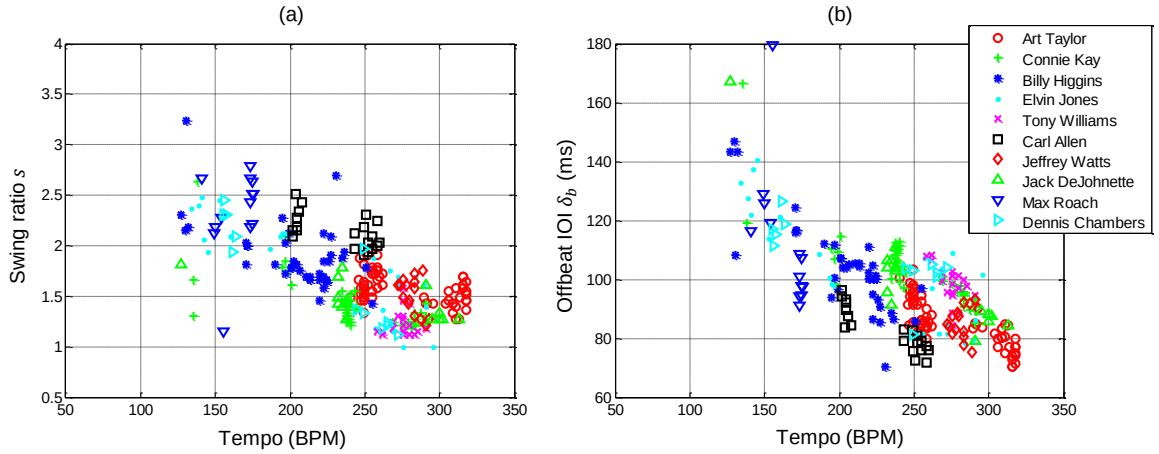

Figure 9.5. Scatter plots showing the relationship of tempo vs. **(a)** swing ratio s and **(b)** offbeat IOI δ_b. Each marker corresponds to one jazz excerpt. We only show the 10 most frequently represented drummers.

while Friberg and Sundström only had around 40 excerpts from various pieces of four drummers, we are able to study several hundreds of RC patterns played by a wide range of drummers due to our automated method (three among them—Tony Williams, Jack DeJohnette, and Jeffrey Watts—were examined by Friberg and Sundström, too).

In Figure 9.5, we show the results obtained for the 10 drummers represented with the most excerpts. Each point in the scatterplots is placed according to (a) s vs. t_e and (b) δ_b vs. t_e. In general, the negative correlation of swing ratio and tempo is clearly discernable—for the whole dataset as well as for certain drummers like Elvin Jones or Billy Higgins, who vary their swing ratio from appr. 2.5 around 150 BPM to appr. 1.5 at 250 BPM, and in the case of Jones even to around 1.0 at 300 BPM. However, there are also drummers who seems to keep almost the same swing ratio at different tempi, e.g., Art Taylor or Carl Allen.

Additionally, Friberg and Sundström report the IOI between the offbeat impulse and the next beat to be roughly constant at 100 ms for all tempi faster than 150 BPM (cf. [81, p. 337]). In general, this finding is supported by our data (see Figure 9.5b), but the offbeat IOIs have a wider range from 110 ms to 80 ms and even 70 ms.

9.5 Conclusions and Further Notes

In this chapter, we presented a microtiming study conducted on a subset of the publicly available WJD. Future work will be directed towards extending our method to more drummers and other recordings as well as to the comparison between RC patterns and soloists. Exact onset times of all tones of the soloists, and thus their microtiming and swing ratio, are at hand within the WJD. A comparison between drummers' and soloists' microtiming will allow for a larger scale re-examination of one of the central findings in [81]: The swing ratio of soloists is in general

lower then the swing ratio of the accompanying drummer since soloists deliberately play behind the beat while synchronizing the offbeat with the drummer. They do so, because, as Friberg and Sundström claim, "delayed downbeats and synchronized off-beats may create both the impression of the laid-back soloist, which is often strived for in jazz, and at the same time an impression of good synchronization" [81, p. 345]. Therefore, using microtiming data from the WJD as well as automatically estimated swing ratios of RC patterns may lead to new insights in the interactive art of improvising together in a professional jazz ensemble.

Chapter 10

The Swingogram

The work in this chapter is mainly based on our contribution in [48].

A typical micro-rhythmic trait of jazz performances is their "swing feel." According to several studies, uneven eighth-notes contribute decisively to this perceived quality. In this chapter we analyze the swing ratio (beat-upbeat ratio) implied by the drummer on the ride cymbal. Extending previous work, we propose a new method for semi-automatic swing ratio estimation based on pattern recognition in onset sequences. As a main contribution, we introduce a novel time-swing ratio representation called swingogram, which locally captures information related to the swing ratio over time. Based on this representation, we propose to track the most plausible trajectory of the swing ratio of the ride cymbal pattern over time via dynamic programming. We show how this kind of visualization leads to interesting insights into the peculiarities of jazz musicians improvising together.

10.1 Introduction

Rhythm and meter constitute essential building blocks of music and are perhaps some of the most accessible aspects for non-expert listeners. The metrical framework that is realized through regular rhythmic structure often induces a motor response of the listener to the music, e. g., tapping one's feet or nodding one's head. Additionally, rhythmic and micro-rhythmic structures contribute to a specific character of the music that humans describe as "swinging," "driving" or "groovy" [35]. In this chapter, we seek to automatically analyze micro-rhythmic variations in the course of recorded jazz solos.

While jazz drummers often emphasize accents of the compositions during the opening and closing thematic sections of a jazz recording, they usually keep time during the solo sections using the

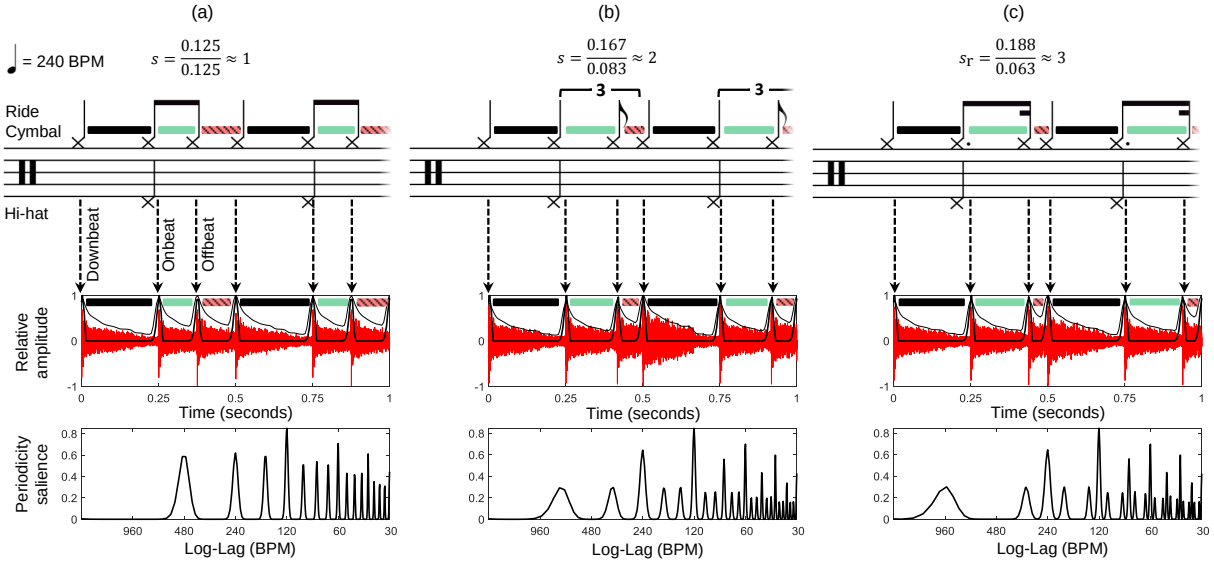

Figure 10.1. Illustration of prototypical RC patterns as drum notation (top), time-domain signal (mid), and LLACF (bottom). Besides the black quarter notes, the relevant eighth-notes are coded by light green (onbeat), and hatched, light red (offbeat). **(a)** Swing ratio of $s = 1$ corresponding to straight eighth-notes, i. e., onbeat and offbeat having the same inter-onset-interval. **(b)** Swing ratio of $s = 2$ corresponding to the idealized tied-triplet notation. **(c)** Swing ratio $s = 3$, where the onbeat can be notated as a dotted eighth-note and the offbeat as a sixteenth-note.

ride cymbal (RC) and hi-hat (HH). Thereby, the RC is struck on every *beat* (i. e., quarter note) while the HH pedal is played on every second beat, i. e., beats two and four in 4/4 bar. Instead of playing the RC steadily, intricate variations and additional *offbeat* strokes are usually intertwined on the RC as well as on other drum parts, especially in styles with so-called *swing feel* [16].
The most common, prototypical RC pattern is depicted in Figure 10.1. In addition to the drum notation in the top row, a corresponding time-domain signal at 240 BPM with overlaid amplitude envelope (thin black curve) as well as an onset-related novelty function (bold black curve) are shown in the middle row. In this figure, the RC onsets, as well as their inter-onset-intervals (IOIs), are color-coded as follows. The sequence starts with a *downbeat* quarter note (black), followed by an *onbeat* eighth-note (light green), and an *offbeat* eighth-note (hatched, light red). This three-note pattern is then repeated. We will re-use these color-codes throughout the chapter, with black indicating any note object that is not considered for measuring swing ratios.
Within jazz performances the musicians deliberately introduce micro-rhythmic variations to convey the typical character of the music. Jazz drummers often play *swinging* eighth-notes, i. e., they modify the beat subdivision and phrasing of the eighth-notes in the RC pattern. Swinging eighth-notes are typically performed in different ratios, on a continuous scale ranging from straight eighths (1 : 1), over tied-triplet eighths (2 : 1), to dotted eighths (3 : 1), and occasionally even more extreme ratios. For example, in Figure 10.1a-c, the color-coded rectangles depict how the onbeat IOI (the time interval between onbeat and subsequent offbeat) grows with increasing

swing ratio. In contrast, the offbeat IOI (the time interval between offbeat and next beat) shrinks. In Figure 10.1a, onbeats and offbeats have equal IOIs, corresponding to straight eighths as typically specified in drum notation. In Figure 10.1b, the swinging eighth-notes are notated as tied-triplets. In Figure 10.1c, the onbeat IOI equals a dotted eighth, resulting in an offbeat IOI corresponding to a sixteenth-note.
According to many authors, the swing ratio generally depends on the tempo (cf. Section 10.2.1). Few works specifically examine the RC pattern [113] and its interaction with the soloist's microtiming [81]. In order to accumulate more data on micro-rhythm in jazz recordings, we introduced a semi-automatic method for swing-ratio estimation from RC patterns [44]. As a main contribution of the current study, we propose an extension to our previous method by introducing a novel time-swing ratio representation that captures the time-varying characteristics of the swing ratio. We apply this method to provide intuitive visualizations of micro-rhythmic variations throughout a jazz performance. We demonstrate the potential of our new method by introducing a tool that can be used by musicologists for analyzing RC swing ratios in relation to tempo, jazz style, and personal preferences of drummers within the history of jazz.
The remainder of this chapter is structured as follows. In Section 10.2, we discuss related work dealing with swing analysis and extraction of rhythmic features in general. In Section 10.3, we introduce our novel *swingogram* representation for visualizing swing-ratio characteristics over time. In particular, we explain the most important swingogram properties and provide details on our proposed extraction strategy. In Section 10.4, we evaluate the robustness of our approach for swing-ratio estimation from jazz solo recordings. Then, in Section 10.5, we apply the swingogram method to point out some preliminary observations about the micro-rhythmic interplay of soloists and drummers in interesting jazz solo recordings. Finally, we conclude and outline important directions for future work in Section 10.6.

10.2 Related Work

In the following two sections, we provide a brief overview of related work that is relevant for our study. Since our research is positioned at the intersection of jazz research and music information retrieval (MIR), we try to cover both aspects. We first discuss some papers with systematic studies on swing ratio in jazz music and then briefly summarize MIR methods that have been proposed for rhythm pattern analysis.

10.2.1 Jazz Microtiming Analysis

An early attempt to analyze swing ratios in jazz solos is described by Kerschbaumer [118]. The author relies on visual inspection of spectrograms but does not report quantitative results.

According to Reinholdsson [170], swing ratios in the analyzed jazz solos range from 1.48 to 1.82. Rose [175] reports an average swing ratio of 2.38 measured from amplitude envelopes of the RC. Ellis [61] measured an average swing ratio of 1.75 using a MIDI wind controller played by saxophonists. Parsons and Cholakis [157] focus on the RC and report swing ratios between 1.0 and 3.3 without detailing the measurement method. Collier and Collier [32] measured an average swing ratio of 1.6 by inspecting amplitude envelopes of soloists recordings. Busse [22] measured an average swing ratio of 2.45 in the performances of pianists playing a MIDI piano. A more comprehensive overview on scientific studies about swing is given by Pfleiderer [163] and Wesolowski [218]. In the following paragraphs, we focus on publications that are closely related to our current study.

Fundamental to our work is the study by Friberg and Sundström [81], who investigated the swing ratio between the onbeat and offbeat in RC patterns by annotating spectrogram excerpts. Their results indicate a linear, negative correlation between the tempo and the swing ratio, which seems to be valid across various drummers. At comparatively slow tempi, the swing ratio reaches up to 3.5, as opposed to fast tempi where it decreases to 1.0. Furthermore, the authors argue that the minimum IOI of the offbeat in RC patterns is around 100 ms, suggesting that there exists an upper limit to the swing ratio achievable at high tempi. In addition, they provide some evidence that the soloists usually play behind the beat of the rhythm section, and perform at a lower swing ratio than the drummer in order to synchronize their offbeats with those of the RC.

Benadon [10] measured comparably low swing ratios up to 1.7 from amplitude envelopes. Additionally, Benadon states that some specific melodic and harmonic features as well as phrase structure and rhythmic peculiarities are elucidated by micro-rhythmic characteristics. Among others, he found a higher swing ratio at phrase endings. Benadon hypothesizes that at these endings the soloist tries to better synchronize to the drummer's RC patterns.

Honing and de Haas [113] conducted experiments with professional jazz drummers performing on a MIDI drum kit. Besides further evidence for the tempo dependency of swing ratios, their results show that jazz drummers have very exact control over their timing. However, they do not conclude that the swing ratio is a linearly decreasing function of tempo. Similarly, Marchand and Peeters [140] report that the swing ratio does not exhibit a linear relationship at tempi lower than 130 BPM and instead shows a preference for the tied-triplet swing ratio around 2.0. Their results are based on manual annotation of downbeat, beat and eighth-note positions for all recordings contained in the GTZAN corpus [195]. However, their study does not specifically focus on jazz performances so the results have to be interpreted carefully.

Davies et al. [35] studied the perception of micro-rhythmic variations. Synthesized rhythm patterns with systematic timing deviations were presented to different groups of listeners who were asked to rate naturalness and liking. Interestingly, only jazz patterns received high ratings when systematic microtiming was introduced by means of swinging eighth-notes.

The majority of the studies mentioned above have in common a two-stage procedure. In the

first stage, a manual annotation of the recordings under analysis is performed, which is a labor-intensive process. In the second stage, an interpretation of the annotated data is devised and, occasionally, compared with listeners responses. When considering large-scale studies, it is important to automate the first step as much as possible. Some researchers have tried to solve this issue by equipping jazz musicians with MIDI instruments [22, 61, 113] and asking them to play under more or less artificial conditions.
In our work, we aim to provide tools that alleviate the annotation process in a semi-automatic manner, thus generating a much larger data sample as was available so far. Furthermore, to better handle large amounts of data, we provide intuitive visualizations via our novel swingogram that immediately exhibits peculiarities of swing-ratio variations within jazz recordings. In Section 10.5, we will demonstrate with concrete examples how studies of micro-rhythm may benefit from the new method.

10.2.2 Rhythmic Mid-Level Features

In MIR, a central paradigm is to convert digitized music recordings into suitable feature representations enabling pattern recognition and retrieval in extensive music corpora. Typically, the first step is to extract "low-level features" which can be computed directly from the audio waveform or equivalent time-frequency representations, such as the short-time Fourier transform (STFT). Besides simple amplitude envelopes [50], many different features have been proposed that emphasize transient events in the signal. For example, spectral flux [49] and novelty curves [146] are commonly used in rhythm analysis. More recently, several authors advocate to use learned feature representations that are optimized for beat or downbeat tracking by means of training deep neural networks [19, 69].
Avoiding the explicit detection of note onset events, time-series of such low-level features can be directly analyzed with regard to re-occurring or quasi-periodic peak patterns. In general, two different families of methods are established for measuring periodicities. Techniques based on the autocorrelation function (ACF) [123, 140] allow to detect periodic self-similarities by comparing a time-series with time-shifted copies of itself. Alternatively, Fourier-based methods such as the beat spectrum [79, 161] compare a time-series with sinusoidal templates representing specific frequencies. Both methods reveal periodicities and local self-similarities, which in turn are the most important cues for rhythmic pattern analysis. For example, the beat spectrum exhibits peaks at frequencies corresponding to the strongest periodicity and its integer multiples. In contrast, the ACF exhibits peaks at integer factors of the strongest periodicities. In an analogy to pitch analysis, these series of peaks are commonly called harmonics, respectively subharmonics. Since it is often difficult to determine which peak corresponds to the rhythmic level of interest (e. g., the beat), one uses the notion of octave ambiguity. To avoid such uncertainties, Kurth

et al. [124] proposed a cyclic post-processing to accumulate rhythmic patterns across all octaves. Both the ACF and the beat spectrum are often referred to as "mid-level features", expressing that they reside on a higher semantic level than the underlying low-level time-series. Ideally, mid-level representations should capture relevant characteristics and remain invariant to aspects irrelevant for the given analysis task. For example, certain rhythmic patterns may be perceived as similar by human listeners even when they are played in a different tempo. Unfortunately, both beat spectrum and ACF are clearly not invariant to tempo differences. As an example, increasing the tempo leads to a compression of the lag-axis underlying the ACF. To counter these problems, several authors proposed conceptually similar post-processing techniques.
Both Dixon et al. [50] and Peeters [161] proposed to warp mid-level features to a common axis normalized to the given bar length. Once converted to this tempo-normalized representation, rhythmic patterns can be compared by simple distance measures, or further rhythmic features can be derived. For example, Marchand and Peeters [140] employed a tempo-normalized peak-fitting model of the ACF for swing ratio estimation.
In order to avoid tempo-informed normalization, Holzapfel and Stylianou [111, 112] proposed to apply the scale transform to ACF-based mid-level features. In theory, the scale transform magnitude of the same rhythmic pattern played in different tempi should be similar up to a constant scaling factor. In practice, the scale transform leads to a new feature representation that is less susceptible to tempo differences. Marchand and Peeters [139] applied the scale transform to modulation spectra, yielding tempo-independent features for the classification of ballroom dances. Similarly, Prockup et al. [167] used the scale transform to derive further descriptors used for the classification of ballroom styles and microtiming characteristics such as swing and shuffle.
The log-lag autocorrelation (LLACF), one of the main ingredients for our method (see Section 10.3.5), was introduced by Gruhne and Dittmar [100]. Around the same time, Jensen et al. [114] proposed a similar, tempo-insensitive representation for ballroom dance classification. In essence, both techniques are based on warping the linearly-spaced lag-axis of the ACF to a logarithmic spacing. The rationale is convert compression (faster tempo) or stretching (slower tempo) of the lag-axis into constant offsets along the log-lag axis. Völkel et al. [214] reported that the LLACF is favorable over the scale transform for classification of Latin American rhythm patterns. Eppler et al. [64] used peak ratios in the LLACF as features for detecting the presence of swing. In our own previous work [44], we proposed to employ LLACF-based pattern matching to estimate the swing ratio of jazz excerpts. In the following section, the main concepts of our previous Chapter 9 will be recapitulated for the sake of clarity.

10.3 Swingogram Representation

In order to semi-automatically track the temporal variation of the drummer's swing ratio in jazz recordings, we introduce suitable features that capture relevant properties while suppressing

irrelevant or confounding information. Given a jazz recording, our core idea is to first extract sequences of LLACF vectors [44] in a segment-wise fashion. These sequences are then converted into a two-dimensional representation that indicates the likelihood of certain swing ratios for each time position. As we will show, our novel representation is, to a certain degree, invariant to tempo changes. Moreover, it can be explicitly interpreted like a spectrogram, with a swing-ratio axis instead of a frequency axis. In reference to the music genre of this study and acknowledging the history of well-known signal representations, such as spectrograms, scalograms [171], chromagrams [6], rhythmograms [115], or tempograms [97], we call our novel mid-level representation swingogram.

10.3.1 Illustrative Example of the Swingogram

First, let us take a look at the instructive example in Figure 10.2 to illustrate the most important properties of the swingogram. We show different signal representations of a synthetic music example that contains 100 repetitions of the RC pattern from Figure 10.1b (tied-triplet feel). Throughout this signal, we steadily increase the tempo with every four beats, yielding a tempo sweep between 146 BPM and 328 BPM. We aim to show that the swing-related properties of the synthetic music signal are contained in each of the three representations, but are most readable in the swingogram. We recommend to take a look at a larger version of Figure 10.2 on our accompanying webpage[40] where the music signal can be auditioned in synchronicity with a cursor showing the current playback position inside each of the signal representations.

In Figure 10.2a, we show a well-known time-tempo representation called *tempogram* [97]. The tempogram is obtained by segment-wise processing of a novelty curve (see Section 10.3.5) with respect to local periodic patterns via STFT. The darker shades of gray encode salient periodicities, while the red curve with white dashes shows the underlying tempo trajectory used in the synthesized signal. To the right side of the tempogram, we mark the position of all integer multiples of the final tempo by black crosses. The second tempo harmonic (third black cross from the bottom up) exhibits the highest peak since it best explains the virtual subdivision of the beat that is inherent to the tied-triplet feel. Note that additional, weaker peaks appear in between the tempo multiples (visible as ridges of lighter shade of gray). These are explained by the periodicity of half the bar-length, i. e., when the RC pattern repeats. Since the tempo changes over time, the tempogram exhibits the same pattern stretched in vertical direction with increasing tempo.

Similarly, in Figure 10.2b, we show the sequence of segment-wise LLACF vectors [44]. For the moment, it is important to note that with LLACFs, the stretching observed in Figure 10.2a transforms into a linear shift along the vertical axis in Figure 10.2b. This is underlined by the

[40] `https://www.audiolabs-erlangen.de/resources/MIR/2017-JNMR-SwingRatio`, last accessed June 14, 2018

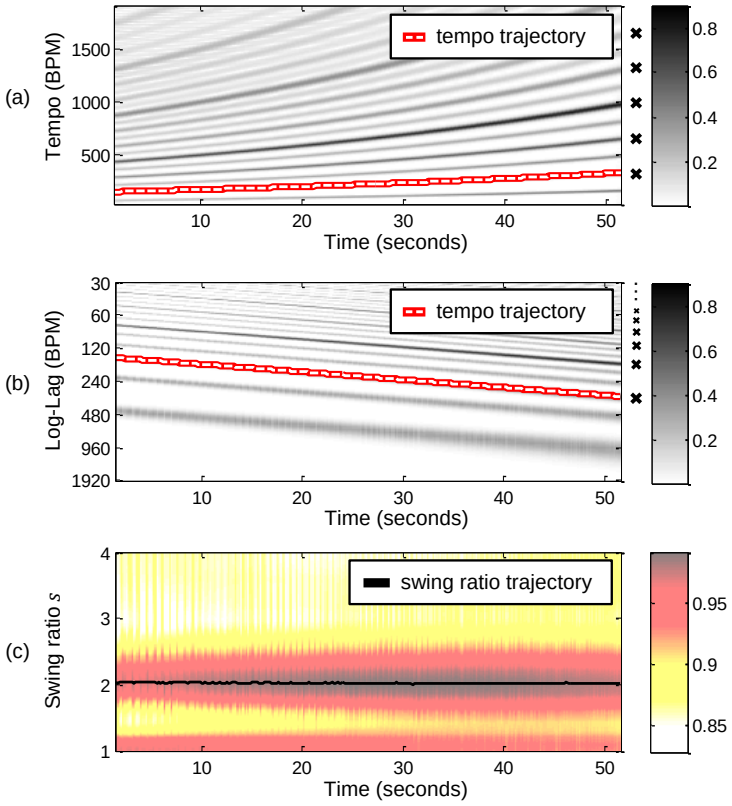

Figure 10.2. Different representations of an RC pattern played with increasing tempo. **(a)** The STFT-based tempogram as described by Grosche and Müller [97]. **(b)** The segment-wise sequence of LLACF patterns (see Section 10.3.5). **(c)** The corresponding swingogram representation.

fact that the tempo trajectory now exhibits a linear slope. Furthermore, note that the vertical axis is reversed w.r.t. Figure 10.2a for the sake of consistency with prior work. We again place black crosses to the right of the LLACF sequence, this time marking the position of the final tempo and its integer factors (tempo subharmonics). Weaker peaks between these tempo values are a result of the characteristic swing pattern. We will discuss the peculiarities of the LLACF and the interpretation of its log-lag axis in more detail in Section 10.3.5.

Finally, in Figure 10.2c, we show the swingogram extracted from the LLACF representation of Figure 10.2b. To emphasize that the swingogram is in a different domain than the previous representations, we encode the values of its elements by a different colormap. Light yellow areas correspond to low likelihood for a certain swing ratio, whereas dark red areas correspond to high values. The black line shows the swing-ratio trajectory that can be tracked via dynamic programming (DP) (see Section 10.3.9). As expected, the trajectory essentially follows the value 2.0 (tied-triplet feel) over the complete duration of our example signal. This clearly shows that the swingogram, as desired, is largely invariant to the tempo changes. In our example, the tempo sweep covers 182 BPM starting with 146 BPM and ending with 328 BPM. Still, the underlying swing ratio can be estimated quite accurately.

10.3.2 Swing Ratio

Before we move on with the details of the proposed extraction method, we now give a proper definition of the swing ratio as used in this chapter. Let $\delta_a \in \mathbb{R}_{>0}$ be the onbeat IOI and $\delta_b \in \mathbb{R}_{>0}$ be the offbeat IOI, i. e., the physical time intervals in the RC pattern as shown in Figure 10.1. The sum of both IOIs equals the beat periodicity denoted by $\delta_{a+b} \in \mathbb{R}_{>0}$. Then, the swing ratio is defined by

$$s = \frac{\delta_a}{\delta_b} \in \mathbb{R}_{>0}. \tag{10.1}$$

In the case that the swing ratio and beat periodicity are known, one can compute the onbeat and offbeat IOIs via the formulas $\delta_a = \delta_{a+b} \cdot s \cdot (1+s)^{-1}$ and $\delta_b = \delta_{a+b} \cdot (1+s)^{-1}$.
Following earlier works, we assume that in jazz recordings, reasonable swing ratios can take any value in the range between $s = 1.0$ and $s = 4.0$. Formally, we express this by introducing the set of possible swing ratios $\mathcal{S} \subseteq [1, 4]$, with $[1, 4] := \{s \in \mathbb{R} | 1.0 \leq s \leq 4.0\}$. In practice, we sample a number $M \in \mathbb{N}$ of discrete prototype swing ratios from $[1, 4]$, assuming that $|\mathcal{S}| = M$. This will be explained in more detail in Section 10.3.6.

10.3.3 Extraction Procedure

Guided by Figure 10.3, we now give a brief overview of our proposed extraction procedure. We refer to the following subsections for details on the individual processing steps.
In short, we propose to first extract a novelty curve (see Section 10.3.4) from the music waveform (Figure 10.3a). Subsequently, segments of the novelty curve are converted into LLACF vectors (see Section 10.3.5). As indicated by the gray boxes in Figure 10.3c, each LLACF vector is compared against LLACF prototype patterns (see Section 10.3.6) by means of a similarity measure (see Section 10.3.8). Finally, in Figure 10.3d the resulting similarity scores yield the elements of a swingogram matrix (see Section 10.3.7).

10.3.4 Novelty Curve

In Figure 10.3a, the waveform of a jazz music excerpt is shown in blue. In black, we overlay the corresponding *novelty curve*, i. e., a time-series that exhibits salient peaks at the temporal position of onset candidates [98]. Given a music signal waveform, the first step for extracting the novelty curve is to convert the signal to an equivalent time-frequency representation using the STFT. Following our previous Chapter 9, the STFT uses a window size of approximately 46 ms and a hop size of approximately 6 ms. By applying logarithmic compression to the STFT magnitude and subsequently accumulating the positive spectral changes between consecutive

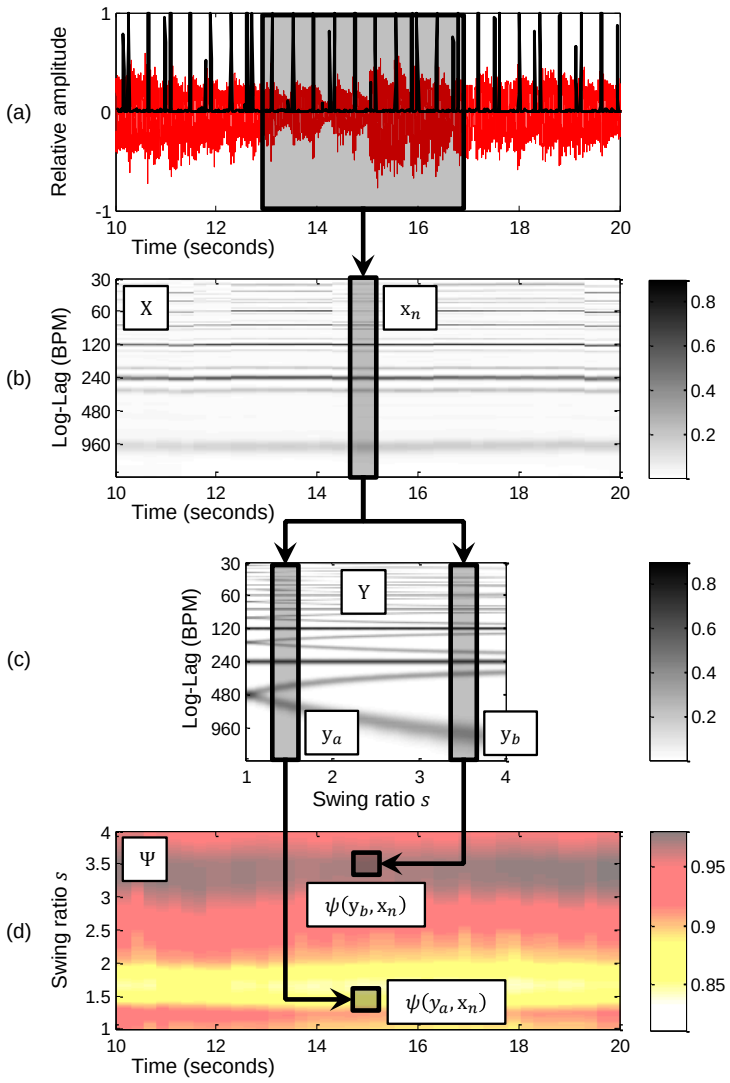

Figure 10.3. Overview of the proposed procedure for computing the swingogram. **(a)** Input waveform of a jazz music excerpt (in blue), overlayed with the onset-related novelty curve (black curve). **(b)** The sequence X of LLACFs extracted in a segment-wise fashion from the novelty curve. The salience of the periodicity peaks is encoded in the gray scale. **(c)** The set Y of LLACF prototype patterns. The salience of the periodicity peaks is encoded in the gray scale. Note that the horizontal axis refers to the range of considered swing ratios. **(d)** The resulting swingogram Ψ after computing the similarity score as described in Section 10.3.7.

frames, one obtains the novelty curve. Peak positions of this curve indicate note onset candidates, see also chapter 6 of the book [146] for details.

RC onsets typically lead to clearly visible transients (i. e., vertical spectral structures) in the upper frequency regions. Thus, we only consider spectral changes in a high-pass frequency band, at the same time attenuating spurious peaks from other instruments. In our previous Chapter 9, we evaluated the suitability of this novelty curve for the detection of RC onsets

in jazz excerpts. Testing against 834 RC onsets annotated by human experts, we received an F-measure of approximately 0.93 using a tolerance of ±30 ms around the true positions. However, especially with older jazz recordings, strongly attenuated high frequency content or crackling and other distortion might lead to severe degradation. Here, more sophisticated extraction strategies based on drum transcription techniques may be applied [37, 174, 189, 213, 220].

10.3.5 Log-Lag Autocorrelation Function

As indicated by the gray box in Figure 10.3a, we extract LLACF vectors from overlapping frames of the novelty curve. Throughout this chapter, we use a frame size of approximately 4 s in conjunction with a hop size of 250 ms between consecutive frames. In essence, LLACF vectors capture the salience of localized periodic repetitions apparent in the novelty curve. For our task, the LLACF serves as a tempo-normalized mid-level representation for rhythmic patterns [44, 100]. To compute the LLACF vectors, we first apply the conventional autocorrelation function (ACF) of the novelty curve inside each frame. Afterward, we use linear interpolation to warp the ACF onto a log-lag axis. This new axis is defined in a way that exhibits equal spacing between tempo octaves and a reference tempo at a defined position.
Let us explain in more detail how we interpret the log-lag axis in BPM. In general, a lag value $\ell \in \mathbb{R}_{>0}$ (in seconds) can be converted to a tempo value $t \in \mathbb{R}_{>0}$ (in BPM) as

$$t = 60 \cdot \frac{1}{\ell}. \tag{10.2}$$

The reciprocal relationship between tempo and lag turns into a negative relationship when applying the binary logarithm

$$\log_2(t) = \log_2(60) - \log_2(\ell). \tag{10.3}$$

The rationale behind using the binary logarithm is to yield unit spacing between tempo octaves, i. e., tempo t and twice the tempo $2 \cdot t$. Formally, this can be expressed as

$$\frac{\log_2(2 \cdot t_a) - \log_2(t_a)}{\log_2(2 \cdot t_b) - \log_2(t_b)} = \frac{1}{1}, \tag{10.4}$$

for arbitrary tempo values $t_a, t_b \in \mathbb{R}_{>0}$. As a consequence, log-tempo and log-lag axis can be interpreted as two sides of the same coin. Once we warp either tempo or lag to binary logarithmic spacing, we can easily switch between both by applying a sign flip (and an offset). This explains why we give the log-lag axis in BPM but in reverse order to the usual orientation.
For the following extraction steps, let $\mathrm{x} \in \mathbb{R}^{K \times 1}$ be an LLACF vector, with $K \in \mathbb{N}$ being the number of elements of the discrete log-lag axis. Furthermore, let $\mathrm{X} := (\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_M)$, be a

sequence of LLACF vectors x_n, with $n \in [1:M] := \{1, 2, \ldots, M\}$. In this context, $M \in \mathbb{N}$ is the number of frames, each of which can be assigned a physical center point. Examples of such LLACF sequences X are shown in Figure 10.2b and Figure 10.3b.
Complementary to the mathematical definition, let us take a look at the bottom row of Figure 10.1 to get an intuition about how the typical shape of an LLACF x changes when fixing the tempo but varying the swing ratio. In Figure 10.1, a pronounced peak at 240 BPM can be seen which represents the periodicity of the beat IOI δ_{a+b}. Slightly higher peaks occur at integer factors of this periodicity, corresponding to the tempi $120, 60, 30, \ldots$ BPM. Note that the sequence of onsets in our prototype RC pattern explains why these tempo subharmonics exhibit a stronger periodicity than the beat itself. As already stated in Section 10.1, the RC pattern repeats after a sequence of downbeat, onbeat, and offbeat, i. e., after two quarter notes. Thus, the highest self similarity of the RC pattern is obtained at half the bar-length. Similar phenomena often lead to octave ambiguity in rhythmic mid-level features as discussed in Section 10.2.2.
Regardless of these issues, it is important to note that the beat-related peaks remain fixed with increasing swing ratio, while the peaks corresponding to the eighth-note IOIs at 480 BPM (and their subharmonics) split up in two peaks of lower salience which encode the characteristic relationship between the onbeat IOI δ_a and the offbeat IOI δ_b. For example, in the case of $s = 3$, the left-most peak resides at 960 BPM, which is exactly the periodicity of a sixteenth-note at 240 BPM. In the following, we will refer to these characteristic local maxima as side-lobe peaks. We exploit their relative location as most important cue for estimating the underlying swing ratio.

10.3.6 LLACF Prototype Patterns

As discussed earlier, our swingogram method reflects the likelihood of certain swing ratios being present in an observed LLACF vector x. The estimation of this likelihood is based on pattern matching against LLACF prototype patterns with known swing ratios. These are obtained by sampling M swing ratios $s_m \in [1, 4]$ for $m \in [1:M]$ and creating ideal LLACF prototype patterns for each discrete swing ratio. In the following, we denote these patterns as $\mathrm{y}_m, m \in [1:M]$. They are obtained by applying the processing steps described above to artificial novelty curves representing the RC pattern with a fixed reference tempo (we use 240 BPM throughout this chapter). In Figure 10.4, we illustrate this principle, revisiting the three example LLACFs from Figure 10.1 at swing ratios $s \in \{1, 2, 3\}$. As indicated by the gray boxes in the bottom part of Figure 10.4, the three prototype patterns are only a small subset of all M prototype patterns. In practice, we represent the set of prototype patterns by a matrix $\mathrm{Y} := (\mathrm{y}_1, \mathrm{y}_2, \ldots, \mathrm{y}_M) \in \mathbb{R}^{K \times M}$ containing the prototype LLACFs y_m as its columns. In the bottom of Figure 10.4 we depict the matrix Y, encoding salient peaks of the prototype LLACFs y_m by darker shades of gray. It is important to note that, opposed to the matrix X (see Section 10.3.5), the horizontal axis now encodes the swing ratio and not time. The above mentioned characteristic side-lobe peaks

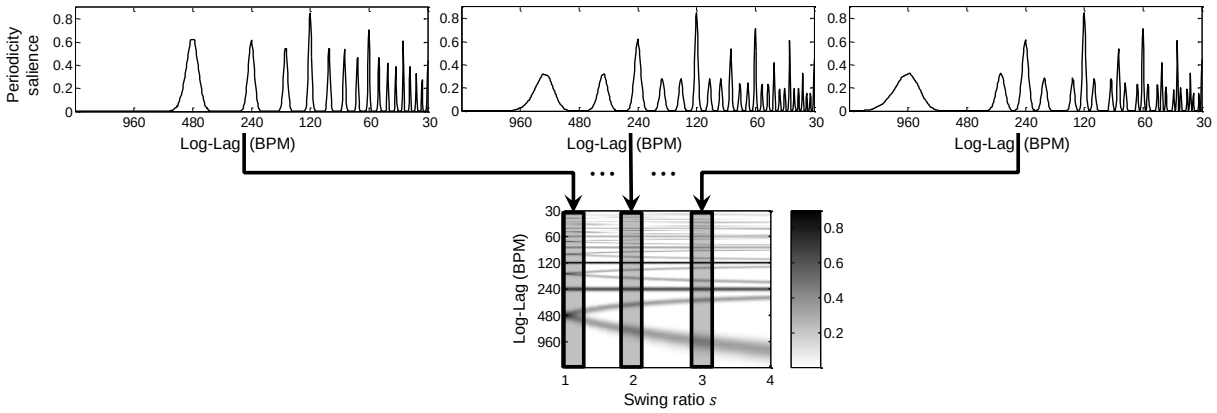

Figure 10.4. Schematic overview of the construction of reference patterns.

induced by the eighth-notes are clearly visible as curved ridges diverging more and more from their initial position (at 480 BPM) with increasing swing ratio s. Note how the same behavior repeats for the multiples of the side-lobe peaks between the stronger peaks corresponding to the beat periodicity δ_{a+b} and its multiples (at $120, 60, 30, \ldots$ BPM).

10.3.7 LLACF Pattern Matching

Returning to Figure 10.3b, we now give an intuitive explanation of our approach to LLACF pattern matching. The slice marked with a dark gray box represents a single LLACF vector x_n extracted from the temporal segment marked in gray in Figure 10.3a. On closer inspection, one might recognize a side-lobe peak around 960 BPM. Recall that we encountered a similar side-lobe peak in the right-most panel in Figure 10.4. This indicates a high likelihood for a swing ratio at $s \approx 3$ but the question remains how we can arrive at this analysis in an automated fashion.
Let us introduce the notion of a similarity function:

$$\psi : \mathbb{R}^{K\times 1} \times \mathbb{R}^{K\times 1} \to \mathbb{R}^1. \tag{10.5}$$

Typically, $\psi(\mathrm{y}, \mathrm{x})$ is high if y and x are similar to each other, and otherwise $\psi(\mathrm{y}, \mathrm{x})$ is small. Evaluating the local similarity score for each pair x_n and y_m, we obtain the swingogram matrix $\Psi \in \mathbb{R}^{M\times M}$ defined by

$$\Psi(m, n) := \psi(\mathrm{y}_m, \mathrm{x}_n), \tag{10.6}$$

for $n \in [1 : M]$ (representing time) and $m \in [1 : M]$ (representing swing ratios).
A conceptual illustration for this procedure is provided by the arrows and boxes connecting Figure 10.3b, Figure 10.3c, and Figure 10.3d. They show how the element x_n of X is compared against two prototype patterns y_a and y_b (columns) of Y. More explicitly, the LLACF prototype

pattern y_a corresponds to $s_a = 1.5$, while the second prototype pattern y_b corresponds to $s_b = 3.5$. By visual comparison of the pattern in x_n with both y_a and y_b, we can see that the LLACF vector under analysis is more similar to the second LLACF prototype pattern. Thus, one obtains $\psi(\mathrm{y}_b, \mathrm{x}_n) > \psi(\mathrm{y}_a, \mathrm{x}_n)$ and the value $\Psi(b, n)$ is larger than $\Psi(a, n)$ as indicated by the color-coding Figure 10.3d.

10.3.8 LLACF Similarity Measure

The notion of likelihood or similarity is of crucial importance in our proposed method. We did not discuss this issue at length before, but we already showed in Figure 10.2b that tempo deviations lead to shifts of the LLACF patterns along the vertical log-lag axis. These potential shifts need to be accounted for when comparing an LLACF vector x to an LLACF prototype pattern y. Therefore, we proposed to use the maximum value of the normalized cross-correlation as a similarity measure [44]. The rationale was to impose invariance to slight tempo differences which might be present due to local deviations from the reference tempo. With slightly simplified notation, our previous similarity score can be expressed as:

$$\psi(\mathrm{y}, \mathrm{x}) := \max_{z} \sum_{k=0}^{K} \mathrm{y}\,(k+z) \cdot \mathrm{x}\,(k)\,, \tag{10.7}$$

for $z \in [-K : K]$, where the vector y is suitably zero-padded. In other words, we only considered the maximum correlation when y is shifted against x.

In this chapter, we introduce an alternative, more efficient method for comparing x and y. To this end, we transform x and y into equivalent representations by computing the complex-valued Discrete Fourier Transform (DFT) and taking the modulus (i. e., absolute value). The rationale is that the above-mentioned log-lag shifts result in phase offsets in the corresponding DFT coefficients. Discarding the phase information makes the similarity score robust against such effects. If we compare an LLACF vector with itself, we want to achieve a similarity score of $\psi(\mathrm{x}, \mathrm{x}) = 1$, thus we normalize the DFT magnitude vectors by subtracting their arithmetic mean and dividing the results by their standard deviation. With regard to our notation introduced above, we call the resulting vectors $\hat{\mathrm{x}}$ and $\hat{\mathrm{y}}$. This transformation has to be computed only once for the fixed LLACF prototype patterns $\hat{\mathrm{y}}_m$. Finally, our similarity score $\psi(\mathrm{y}, \mathrm{x})$ simplifies to an inner product:

$$\psi(\mathrm{y}, \mathrm{x}) := \langle \hat{\mathrm{y}} | \hat{\mathrm{x}} \rangle. \tag{10.8}$$

Clearly, this is a much more streamlined and elegant procedure than the one proposed in our previous work [44].

10.3.9 Swing Ratio Tracking

In order to obtain a trajectory of the swing ratios in Ψ, one could go from left to right through each of the M columns, each time picking the maximum element and looking up the corresponding swing ratio s. However, this might not be robust against sudden jumps. In order to achieve smooth trajectories considering the temporal context, we propose to use *Dynamic Programming* (DP) as a standard technique to find an *optimal path* that maximizes the similarity score among all possible paths in Ψ. The general idea behind DP is to use an accumulated similarity matrix and store local maximum decisions to recursively find the score-maximizing path [60, 146]. Furthermore, constraints can be added on the allowed local slope of the trajectory when going from one column to the next.
For our scenario, we make two assumptions for the application of DP. First, we presume that the drummer will change the swing ratio only gradually from frame to frame. Second, the LLACF prototype patterns encoded by Y are generated according to the continuously increasing scale of swing ratios. The typical outcome of a DP-based tracking is overlayed as a black path on top of swingogram Ψ in Figure 10.2c.

10.4 Evaluation

In this section, we evaluate the quality of swing-ratio estimates derived with our swingogram representation. First, in Section 10.4.1, we give an overview of the manually annotated jazz excerpts we use as ground truth data. Then, in Section 10.4.2 and Section 10.4.3, we present the experimental results from two perspectives.

10.4.1 Dataset and Annotations

To a large extent, our research is driven by the Jazzomat Research Project [82]. The project aims to investigate the creative processes underlying jazz solo improvisations via computational methods. The overarching goal is to explore the cognitive and cultural foundations of jazz solo improvisation. As a basis for our work, we can use the jazz solo transcriptions and music recordings of the Weimar Jazz Database (WJD[41]) that has been created within the Jazzomat project.
The WJD consists of 430 (as of February 2017) transcriptions of instrumental jazz solo recordings performed by a wide range of renowned jazz musicians. These recordings are characterized by

[41] `http://jazzomat.hfm-weimar.de/dbformat/dboverview.html`, last accessed June 14, 2018

a predominant, monophonic solo instrument (e. g., trumpet, saxophone, clarinet, trombone) playing simultaneously with the accompaniment of the rhythm group (e. g., piano, bass, drums). The onsets and offsets of each note played by the soloists have been manually transcribed and verified by musicology and jazz students at the University of Music Franz Liszt Weimar. In addition, the database contains further musical annotations (e. g., beats, chords, phrases) as well as basic meta-data about the jazz recordings (artists, record name, recording year).
For our work, we asked two experienced student assistants involved in the Jazzomat project to transcribe RC onsets in excerpts of 48 solos contained in the WJD. In total, 3945 RC onsets were manually annotated using the software Sonic Visualiser [26]. As we will explain later, approximately 10 % of these onsets have been transcribed redundantly to enable evaluation of the annotator agreement.
In order to derive ground truth swing ratios from the manual onset annotations, a subset of $L = 955$ triples was automatically selected. In our context, the term "triple" [44] refers to an onset sequence of onbeat, offbeat and subsequent onbeat as shown in Figure 10.1. For each of the L triples, the time differences between the consecutive onsets yields the onbeat IOI δ_a and the offbeat IOI δ_b. Consequently, we use equation (10.1) to compute ground truth swing ratios, which we denote as $r_\ell \in [1, 4]$, with $\ell \in [1 : L]$. Note that these triple-based ratios adhere to our swing ratio definition in Section 10.3.2 with $1.0 \leq r_\ell \leq 4.0$. In practice, we achieve this by only accepting onset sequences as triples if they fulfill the following three conditions:

- $\delta_{a+b} \leq \frac{1.2 \cdot 60}{t}$, i. e., the beat IOI must be close to the reciprocal of the given tempo.
- $\delta_b \leq \delta_a$, i. e., the offbeat IOI is expected to be shorter than the onbeat IOI.
- $\delta_a \leq 4.0 \cdot \delta_b$, i. e., the onbeat IOI must not exceed an integer multiple of the offbeat IOI.

In the following, we refer to the combined conditions as "triple criterion".
We can assign to each r_ℓ a time position ν_ℓ (given in seconds) according to the triple center point. Recall that each column of our swingogram Ψ is also assigned a physical time position. Consequently, we obtain for each ground truth r_ℓ the corresponding estimate s_ℓ from the swing-ratio trajectory element corresponding to ν_ℓ. This principle is illustrated in Figure 10.5, where we depict the automatically extracted swing-ratio trajectory as a quasi-continuous curve, while the reference swing ratios are only given at discrete points in time. Using these pre-requisites, we quantify the deviation between the ℓ^{th} pair of the reference swing ratio r_ℓ and estimated swing ratio s_ℓ by their difference as:

$$\varepsilon_\ell := s_\ell - r_\ell. \tag{10.9}$$

In the following, we introduce four conditions covering different combinations of ground-truth swing ratios and swing-ratio estimates. First, for the *swingogram condition* (SG), we use the swing-ratio estimates s_ℓ extracted by our proposed swingogrammethod. Second, we consider

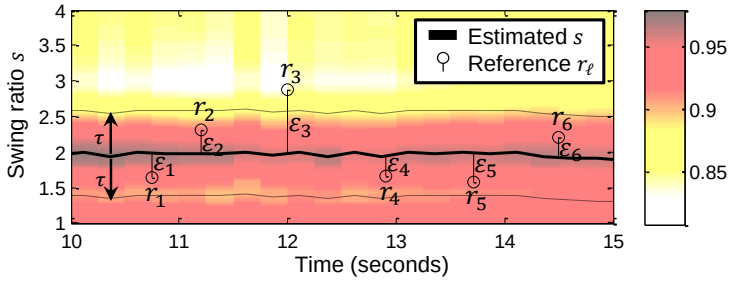

Figure 10.5. Deviation ε_ℓ and tolerance τ that we employ for evaluating our proposed method.

the *lazy guess condition* (LG) that serves as a lower performance bound. For LG, we do not perform any automatic swing-ratio estimation and instead simply assume a constant swing ratio adhering to the tied-triplet feel (i. e., $s_\ell = 2.0$), regardless of the given tempo. At this point, we need to mention that our manually annotated solos exhibit a strong bias for the median tempo of approximately 200 BPM and the median swing ratio of approximately 2. Third, we employ the *educated guess condition* (EG), where the swing ratio is computed as a linear function of the given tempo. This linear model is obtained by fitting a first-order polynomial to the tempo and swing-ratio pairs reported by Friberg and Sundström [81]. Finally, we introduce the *cross-validation condition* (CV) using a subset of $L = 95$ ground truth swing ratios, for which the second annotator provided an independent transcription of the same solo excerpts as the first annotator. In this context, we interpret the swing ratios by the first annotator as ground truth r_ℓ, and consequently, the swing ratios by the second annotator as our estimates s_ℓ. This is to obtain some idea of an upper performance bound.

10.4.2 Evaluation with Accuracy Metric

For our first, accuracy-based evaluation perspective, we introduce a tolerance parameter $\tau \geq 0$ for the maximal acceptable deviation between all pairs of ground truth r_ℓ and estimate s_ℓ. Given ε_ℓ between all corresponding pairs of true and estimated swing ratios, we quantify the accuracy as follows:

$$A_\tau := \left|\{\ell \in [1:L] : -\tau \leq \varepsilon_\ell \leq \tau\}\right| / L. \tag{10.10}$$

We sweep through increasing tolerance values in the range $\tau \in [0, 2]$, expecting to see rising accuracy curves as we make the evaluation less and less strict.

In Figure 10.5, we illustrate the relationship between the quantities introduced above in an example with $L = 6$ ground truth swing ratios measured at discrete points in time. As introduced in Figure 10.2c, the bold black line depicts the swingogram-based swing-ratio trajectory. The black dashed lines running in parallel to the trajectory show the tolerance area whose width is

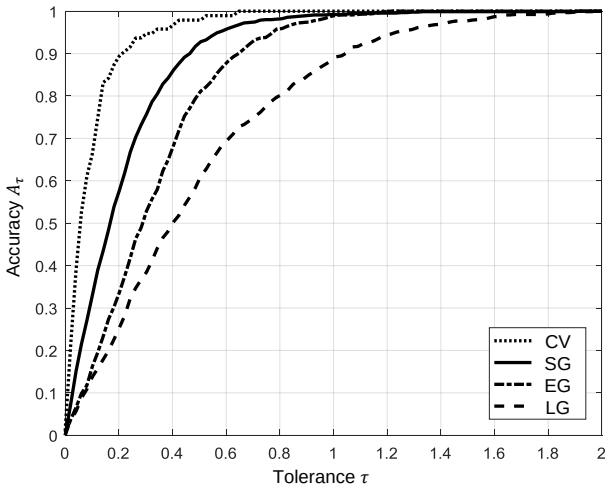

Figure 10.6. Results of our evaluation from the accuracy perspective.

determined by τ. For example, the error ε_3 (centered at $\nu_3 \approx 12$ s) clearly exceeds τ.
In Figure 10.6, the dashed line shows the accuracy for condition LG, while the dash-dotted line shows the accuracy for condition EG. We can observe that the linear swing-ratio model of condition EG already yields better results than the constant model used in condition LG. However, both curves are clearly below the solid black line representing the results that can be achieved with swing ratios extracted via our proposed method (SG). Considering the dependency on the tolerance τ, we see that one can obtain a reasonable accuracy $A_\tau \approx 0.85$ when accepting absolute deviations between estimated and true swing ratios up to $\tau = 0.4$. For the CV condition, we can see that the accuracies approach 0.85 already below $\tau = 0.2$.

10.4.3 Evaluation with Root Mean Squared Error

For the second evaluation perspective, we revert to the model assumption that our automatically estimated s_ℓ can be interpreted as noisy measurements of the true r_ℓ. We use the Root Mean Squared Error (RMSE) as a quality metric:

$$\text{RMSE} := \sqrt{\frac{1}{L}\sum_{\ell=1}^{L} \varepsilon_\ell^2}. \tag{10.11}$$

In general, the RMSE value is unbounded but should be close to zero in case of low error.
In Figure 10.7, the black bars show the RMSE that is achieved for the test conditions LG, EG, and SG (using $L = 955$ ground truth triplets). The highest error of RMSE ≈ 0.62 is apparent for LG, the condition assuming a constant swing ratio $s_\ell = 2$. The second highest error RMSE ≈ 0.4

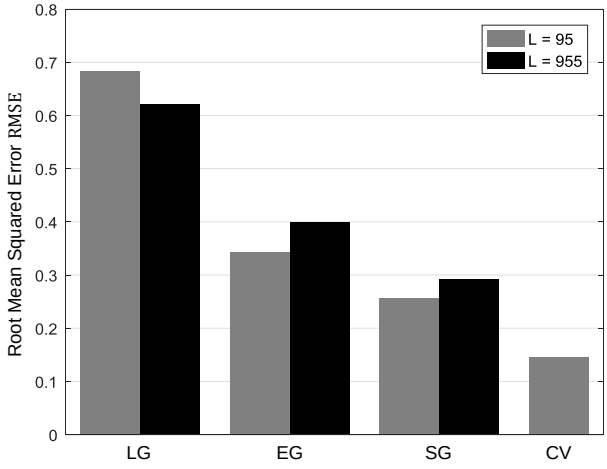

Figure 10.7. Results of our evaluation from the RMSE perspective.

occurs for condition EG, assuming a linear relationship between the given tempo and the swing ratio. In comparison, our swingogram-based estimates (SG) achieve RMSE ≈ 0.3. As expected, the condition CV shows the smallest error with RMSE ≈ 0.15 (in gray, using $L = 95$ ground truth triplets). It is not surprising that human experts show a certain level of disagreement when transcribing RC onsets. Aside from differences in perception, an unknown fraction of this error can also be attributed to technical inaccuracies of the tools used for manual transcription.

Since we only had $L = 95$ swing-ratio pairs available for comparison in the CV case, we re-evaluated the conditions LG, EG, and SG with the same subset of ground truth triplets. In that case, as shown by the gray bars, both EG and SG show a slightly better performance, both error metrics drop by approximately 15 %. In contrast, the RMSE for condition LG even increases. This can be explained by the fact that the ground truth swing-ratios in the CV subset are more evenly distributed and less biased towards the tied-triplet feel. Overall, we can see that our proposed method for swing ratio estimation has an advantage over simpler models, although it is not on par with human performance. It remains to be seen how the general tendencies change once a larger set of manual RC annotations is available.

10.5 Micro-Rhythm Analysis in the Swingogram

In this final section, we want to explore RC swing ratios extracted from recordings of improvisations included in the WJD. First, we re-examine the relationship between tempo and swing ratio. Then, we discuss three excerpts, where preliminary observations about different strategies of micro-rhythmic interplay between soloist and drummer can be made.

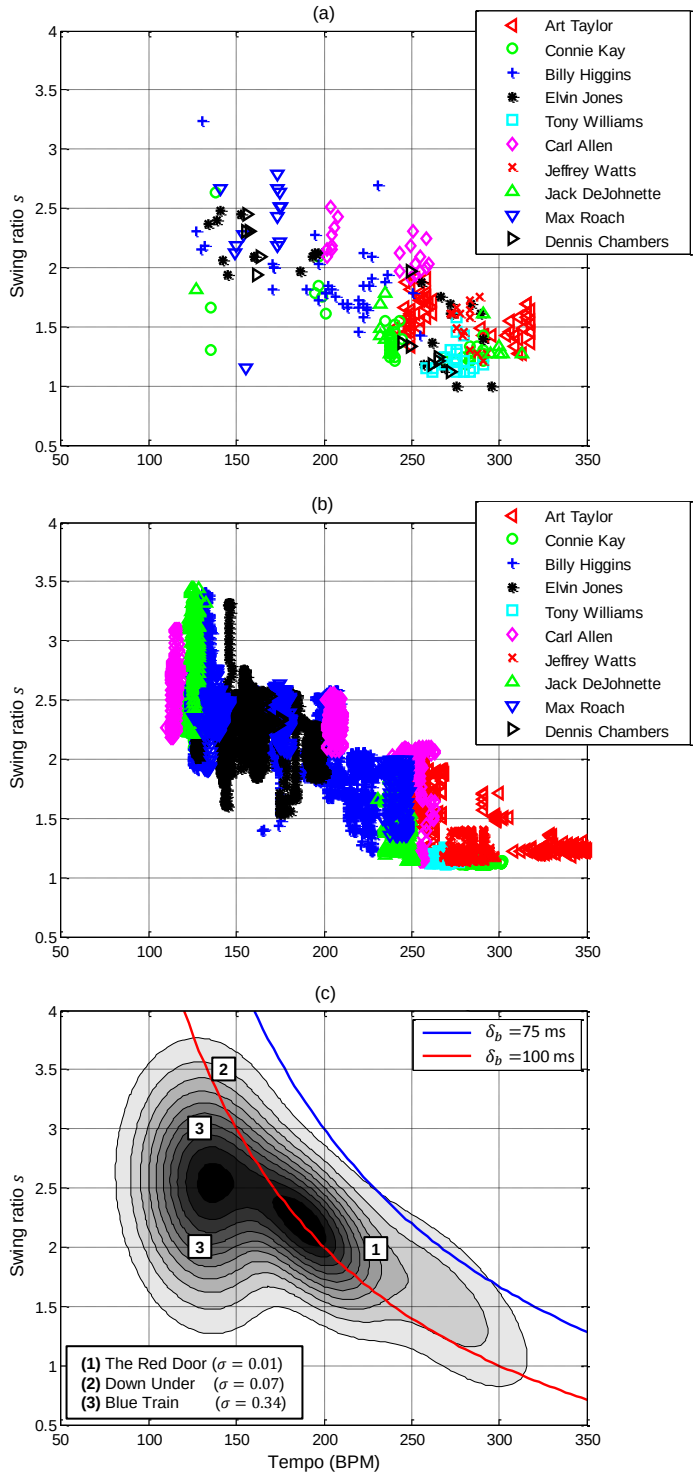

Figure 10.8. Tempo vs. swing ratio in the WJD. **(a)** Scatterplot from our previous Chapter 9. **(b)** Scatterplot with much higher number of tempo and swing-ratio pairs. **(c)** Statistical model of the same data, overlayed with hypothetic swing-ratio limits and pointers to selected examples (see Section 10.5.2.

10.5.1 Tempo vs. Swing Ratio

Continuing our previous work [44], we re-examine the much-debated relationship between tempo and swing ratio of the drummer (see Section 10.2.1). In Figure 10.8a, we show the results from our previous Chapter 9 for reference. Each point in the scatterplot is placed according to swing ratio and tempo estimated in short excerpts of recordings contained in the WJD. The different markers and colors encode the names of renowned jazz drummers who played in the excerpts under analysis. For the sake of visibility, we only depict the 10 drummers that appeared most frequently. In total, 278 excerpts with a typical duration between two and six seconds went into this figure, each of these excerpts corresponds to one pair of tempo and swing ratio. In the current chapter, we can evaluate much larger quantities of swing-ratio estimates, by considering entire trajectories of swing ratios covering almost all relevant solo recordings contained in the WJD.

In Figure 10.8b, we show the scatterplot that we obtain with our swingogram-based extraction of swing-ratio trajectories. Compared to Figure 10.8a, it can be seen that our tempo vs. swing-ratio diagram is much more densely populated with points. A total of 67 solos[42] with swing rhythm feel went into this plot, leading to $28,851$ data points. This large number results from the fact that we extract the swingogram Ψ with a constant hop size of 250 ms between consecutive LLACF segments. The resulting swing-ratio trajectories exhibit the same temporal resolution, yielding up to 240 swing-ratio estimates per minute.

Due to the large amount of data, we decided to approach the tempo vs. swing-ratio question from a statistical modeling perspective for a better overview. To this end, we model the probability of a certain swing ratio appearing at a certain tempo by a Gaussian Mixture Model (GMM) that is fit to the data points of Figure 10.8b. In Figure 10.8c, we show the resulting two-dimensional probability density function as a contour plot, where darker shades of gray encode higher likelihood to observe a certain combination of tempo and swing ratio given the WJD data. We can observe in general that higher tempi lead to lower swing ratios. However, it also becomes evident that this is not a strong negative correlation but rather a wide band of possible swing ratios with increasing spread at lower tempi.

This increased variance is also visible in Figure 10.8b, where there are clusters of data points that are shaped like vertical lines and appear at the same tempo (e. g., between 100 BPM and 150 BPM). Of course, the huge number of data points at a certain tempo are due to the fact that the data is taken from a limited number of recordings each of them having a definite tempo with only tiny variations. Thus, there is no statistical distribution with regard to tempo but, on the contrary, many data points accumulated at certain tempi. Note that many of the columns have different colors according to certain drummers playing on certain recordings each with an almost constant tempo.

[42] The metadata for these solos are provided on our accompanying webpage: `https://www.audiolabs-erlangen.de/resources/MIR/2017-JNMR-SwingRatio`, last accessed June 14, 2018

On the other hand, the columns might also be caused by estimation errors introduced by our proposed method. Recall that we obtained an accuracy $A_\tau \approx 0.95$ when we allowed a tolerance of $\tau \approx 0.5$ (see Figure 10.6). This uncertainty could be an explanation for the observed line-shaped clusters, especially the ones that exhibit a swing ratio spread of ± 0.5 around their center.
Additionally, jazz drummers might deliberately add variation to their swing ratio, especially at lower tempi, in order to make their playing more vivid and to interact with the soloist. To further investigate this phenomenon, we show the average tempo and swing ratio of three jazz excerpts corresponding to the markers in Figure 10.8c. Their titles are given in the lower left legend with the standard deviation of the swing-ratio trajectory given in brackets (denoted as σ). Note that the swing-ratio estimates of these three recordings are not among the data points underlying Figure 10.8b. In Section 10.5.2, we will re-use these three examples to discuss in detail observations about the interaction between soloist and drummer. For now, it is important to note that excerpts (1) (see Section 10.5.3) and (2) (see Section 10.5.4) exhibit rather small variation in the drummer's swing ratio, whereas excerpt (3) shows a considerable spread. As we show in Section 10.5.5, this recording features a change in the rhythmic play of both the soloist and the drummer along with a considerable drop in swing ratio. To account for this bimodal swing ratio behavior, we added two boxes to our figure, each of which is centered at the average swing ratio dominating in the different sections of this excerpt.
Finally, the red line in Figure 10.8c depicts a hypothetical swing ratio that would be caused by sweeping along the tempo axis with a fixed offbeat IOI of $\delta_b = 100$ ms. This line was brought up by Friberg and Sundström [81] as an upper limit to the swing ratio that can be observed in jazz music. For our data, however, the red curve cuts straight through the ridge of highest probability. Already in our previous study, we found many jazz excerpts with lower offbeat IOI, going down to $\delta_b = 70$ ms in some rare cases. Since we manually double-checked these findings and are confident that they are not caused by extraction errors, we propose to shift the border of the hypothetical "no-go area" towards the blue line corresponding to $\delta_b = 75$ ms.

10.5.2 Micro-Rhythmic Interaction between Drummers and Soloists

We close this chapter with three observations of short excerpts taken from the WJD. In Section 10.5.3, Section 10.5.4, and Section 10.5.5, we discuss each excerpt as a case study on how our novel swingogram can help to quickly assess the interaction between drummer and soloist. As explained in Section 10.2.1, we want to re-examine certain observations made by Friberg and Sundström [81] and Benadon [10]. In essence, we are interested in three hypotheses:

1. Soloists tend to play with a lower swing ratio than the accompanying drummers.

2. Soloists try to synchronize their offbeat onsets to those of the drummers.

3. Soloists strive for higher swing ratios and thus better synchronization at phrase endings.

The panels in Figures 10.9,10.10, and 10.11 are arranged in the same fashion. In the top panel (a), we show a swingogram overlayed with the swing-ratio trajectory as introduced in Section 10.3.1. While panel (a) covers the complete solos (duration between 30 and 60 seconds), we additionally mark an excerpt of interest by vertical dashed lines. In panels (b) and (c), we provide a zoomed view into these sections, each of which are three measures long (i. e., 12 beats).
For each solo section, the lower left panel (b) provides both a score notation and a piano-roll notation of the tones played by the soloist. The vertical gray lines in the background visualize the beat grid that we directly obtain from manual annotations in the WJD. The dashed gray lines depict the corresponding offbeat positions which we computed from the automatically extracted swing-ratio trajectory. We arranged the note heads to correspond to the onsets of the piano roll objects. Furthermore, we use the same color-coding for onbeat and offbeat as introduced in Figure 10.1 in order to highlight the interesting eighth-notes of the soloist.
Finally, the lower right panel (c) provides a zoom into the swingogram. In addition to the drummer's swing-ratio trajectory, we again depict the beat grid by vertical gray lines. On top, we indicate the swing-ratio estimates of the soloist by vertical dashed line segments. These estimates are based on the triple criterion (see Section 10.4.1) in combination with the annotations of the metrical position for each note as contained in the WJD. In contrast to the drummer's swing ratio, the soloist swing ratio can only be measured when sequences of eighth-notes are played.
We recommend to listen to the Youtube videos of our examples that we provide on our accompanying webpage.[43] During playback, a synchronized cursor highlights the current position within the swingogram as well as the piano-roll notation of the solos. Some of the phenomena discussed below are clarified by listening to the music examples.

10.5.3 The Red Door

In Figure 10.9, we analyze an excerpt from "The Red Door", recorded by the Gerry Mulligan group in 1960, in order to study the micro-rhythmic interplay of solo baritone saxophonist Gerry Mulligan and drummer Mel Lewis. In Figure 10.9a, it can be observed that Lewis keeps the idealized tied triplet swing ratio $s \approx 2.0$ for more than 60 seconds at an average tempo of 236 BPM. Mulligan plays in the same precise fashion, almost always synchronizing his onbeat onsets to those of Lewis. However, he shows quite some variability in his offbeat onsets, indeed showing higher swing ratio at phrase endings, as hypothesized by Benadon [10].

[43] `https://www.audiolabs-erlangen.de/resources/MIR/2017-JNMR-SwingRatio`, last accessed June 14, 2018

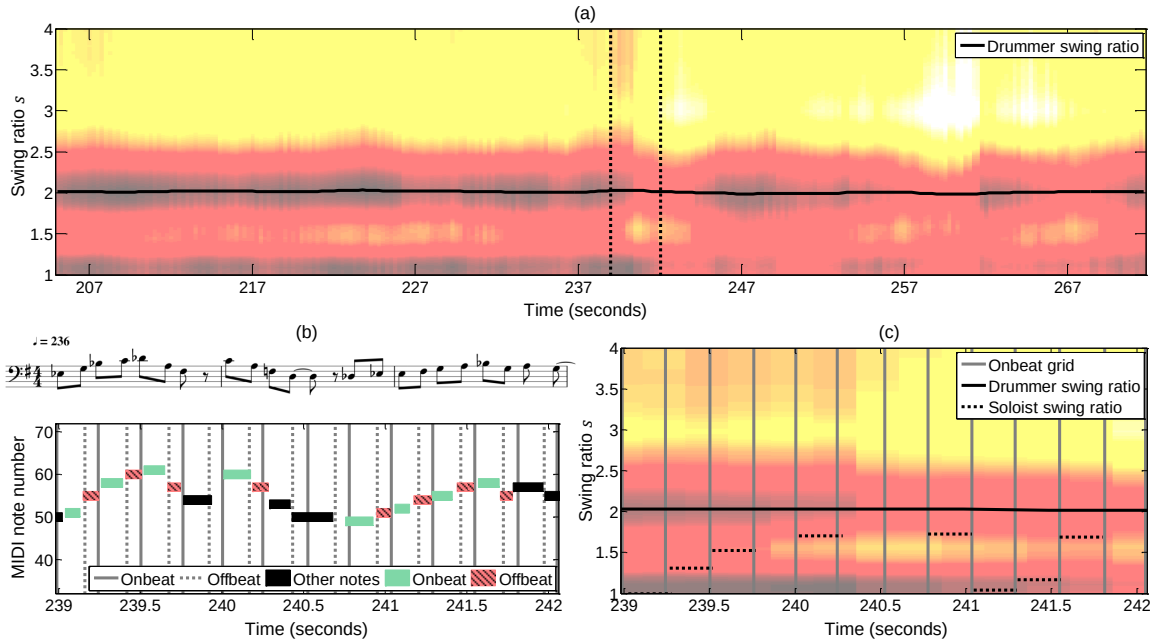

Figure 10.9. Swingogram analysis of a solo-section from the 1960 recording of "The Red Door". See Section 10.5.3 for discussion.

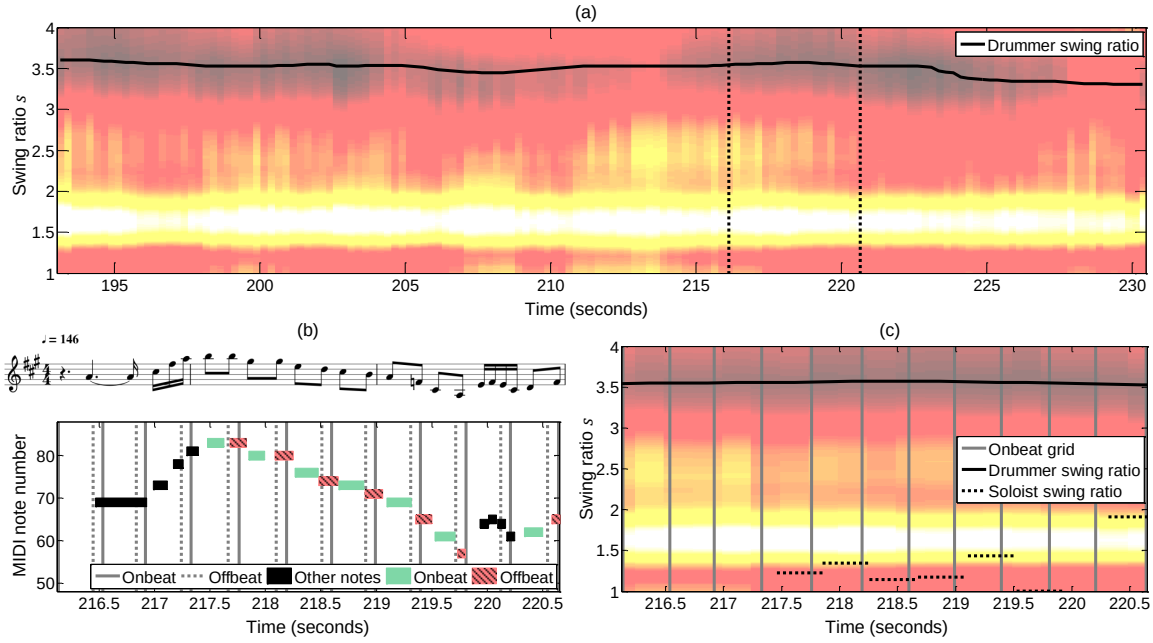

Figure 10.10. Swingogram analysis of a solo-section from the 1961 recording of "Down Under". See Section 10.5.4 for discussion.

10.5.4 Down Under

In Figure 10.10, we show a solo section from "Down Under," recorded by Art Blakey's Jazz Messengers in 1961, with trumpet player Freddie Hubbard performing together with drummer

and bandleader Art Blakey. At an average tempo of 146 BPM, the RC swing ratio exhibits a slightly wavy variation around $s \approx 3.5$. This gives an example of the archetype of drummers playing high swing ratios at low tempi as discussed in Section 10.5.1. In constrast, Hubbard stays at the other end of the swing-ratio range at around $s \approx 1.25$ in the zoomed-in section. Furthermore, one can clearly see that Hubbard plays his onbeat onsets in a laid-back fashion behind the beat grid. His offbeats, however, synchronize very well to Blakey's offbeats most of the time as hypothesized by Friberg and Sundström [81].

10.5.5 Blue Train

In Figure 10.11, we present an excerpt from John Coltrane's recording "Blue Train," (1975) featuring solo trombone player Curtis Fuller and drummer Philly Joe Jones. At an average tempo of 132 BPM, the RC swing ratio starts around $s \approx 3.0$ and drops in the middle part to $s \approx 2.0$. The change in swing ratio is coupled with the start of a completely different drum pattern. In contrast to our previous, well-behaved examples, the RC plays only onbeats (i. e., quarter notes), while the HH is placed on straight offbeats, thus conveying a double-time feel (i. e., the impression of twice the tempo). On closer inspection, we found that swinging eighths notes in the middle part are played solely on the snare. Still, the snare hits lead to spikes in the novelty curve due to crosstalk into the RC frequency band.
Interestingly, both the bassist Paul Chambers and trombonist Curtis Fuller do not follow the rhythm change immediately. This might explain the rather loose interplay during the excerpt where neither the onbeats nor the offbeats played on the trombone synchronize particularly well to the drummer. Instead, the soloist's swing ratio oscillates around the drummer's swing ratio. However, shortly after the end of our zoomed section, Fuller switches to sixteenth-note sequences, supporting the double time feel.

10.6 Conclusions and Further Notes

In this chapter, we introduced the swingogram, a novel time vs. swing-ratio representation suited for analyzing the time-varying behavior of swing ratios in jazz solo recordings. We evaluated the accuracy of our semi-automatic swing-ratio estimates by comparing them against ground-truth annotations using two different evaluation metrics. We revisited the debated linear relationship between tempo and swing ratio that has been in the scope of several researchers. Thanks to our swing-ratio estimation method and the availability of the WJD corpus, we were able to base our analysis on a considerably larger collection than previous studies. This lead to new insights about the probability density distribution of swing ratios in different tempo ranges as well as the hypothetic upper limit to swing ratios. Using three examples from the WJD, we illustrated

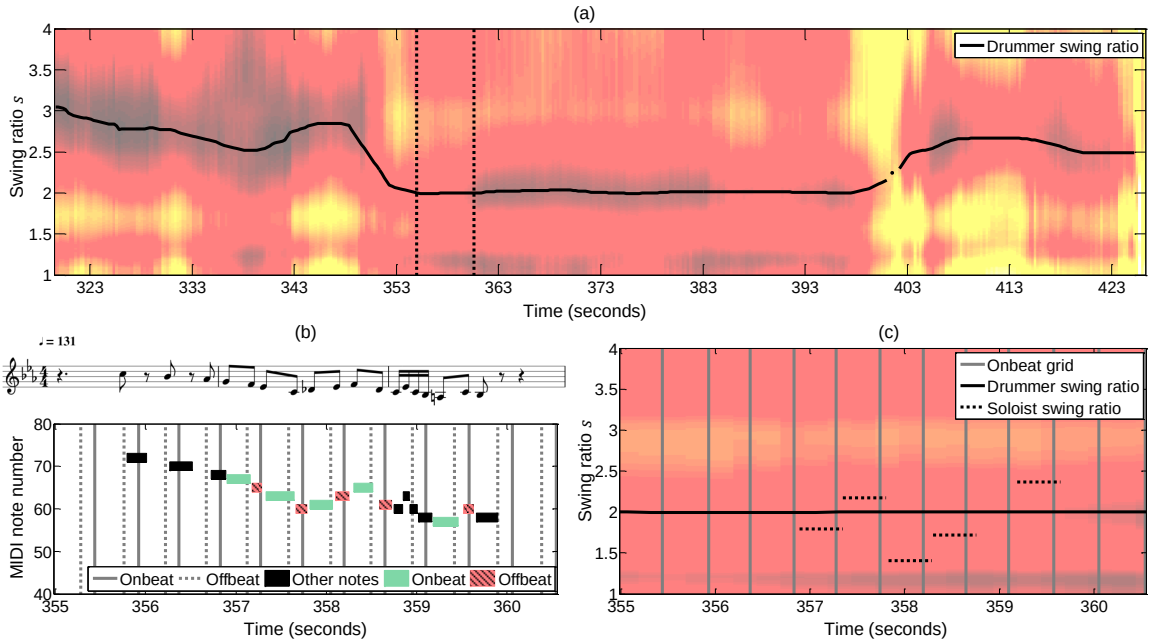

Figure 10.11. Swingogram analysis of a solo-section from the 1957 recording of "Blue Train". See Section 10.5.5 for discussion.

how our swingogram visualization can support the understanding of the interaction between the soloist and the drummer. This immediate access to interesting solo sections is of pivotal interest for a comprehensive analysis of micro-rhythm within a jazz performance.

Future work will be directed towards investigating the interaction between drummer and soloist as discussed by Friberg and Sundström [81] on a larger scale. In principle, sufficient data sources required to allow statistical evaluations of their hypotheses are available. However, in the current state, some important intermediate steps are difficult to automate. The identification of soloist triples in ambiguous cases remains challenging, as well as the automatic rejection of inadequate swing-ratio trajectories in the swingogram.

From the viewpoint of signal processing, we plan to investigate the advantages and disadvantages of the LLACF-based swingogram against other extraction methods such as the scale transform [112] or the shift-ACF [123]. At this point, it seems promising to also apply these methods for the analysis of other micro-rhythmic phenomena, such as shuffle or groove, too.

Chapter 11

Summary and Future Work

In this thesis, we investigated computational techniques and application scenarios for source separation and restoration of drum sounds in digitized music recordings. Although this might seem like a niche topic, it entails some challenging audio processing problems of broader interest. Motivated by those issues, we were able to gain new insights into some under-explored aspects of source separation applied to signals that are of transient and inharmonic nature. In retrospect, it is remarkable that large parts of this thesis orbit around a small number of algorithmic core concepts. Most prominently, Non-Negative Matrix Factorization (NMF) [130] and its convolutional counterpart Non-Negative Matrix Factor Deconvolution (NMFD) [185] were of central importance for our work.

As documented in Chapter 2, a number of papers on Automatic Drum Transcription (ADT) have used NMF and its variants to extract activation functions by means of spectrogram decomposition. Consequently, we compared ADT capabilities of five different NMF-based ADT methods to five other methods that rely on Recurrent Neural Networks (RNN) in Chapter 3. Although the RNN-based approaches generally yield better ADT performance, NMF and NMFD have the advantage that they require very little training data and are well suited for source separation purposes as well. Thus, NMF and NMFD have re-appeared throughout the second part of this thesis, namely in Chapters 4, 5, 6, and 8. In these chapters, we have more closely investigated parameters and constraints that are important for source separation of drum sound events. In Chapter 4 for example, we showed that score-based initialization of NMF activations (i. e., only using onset-related information) performs almost as good as audio-based initialization of NMF templates (i. e., using entire training examples). Moreover, we showed that shaping NMF activations to resemble decaying impulses is appropriate for drum sound extraction, see Chapter 8. In both Chapter 5 and Chapter 4, we used dictionary-based restoration methods to repair imperfect spectrogram decompositions that may result from unwanted convergence of NMF to suboptimal solutions.

Subsequent to spectrogram decomposition, we used Wiener filtering in conjunction with an inverse STFT as central techniques for signal reconstruction. In Chapter 6, we showed that it is generally appropriate to use the phase of the mixture for reconstruction in the context of drum sound separation. Improvements can be obtained by our proposed transient restoration method based on time-domain constrained iterative phase reconstruction. In the same vein, we experimented with generalized Wiener filtering in Chapter 7. Although yielding inconclusive results, this chapter gave rise to our variant of harmonic-percussive source separation (HPSS) using the Kernel Additive Modeling (KAM) paradigm. We re-used the same technique in Chapter 8 as a means to obtain initial estimates of the harmonic and percussive part of music recordings. These estimates are in turn refined by a modification of NMF using novel soft constraints.
In the final part of this thesis, our contributions were oriented towards musicology research about micro-rhythmic phenomena in jazz music. Returning to an ADT-related task in Chapter 9, we evaluated the suitability of ride cymbal (RC) onset detection for estimating the swing ratio in jazz recordings. We proposed an alternative method that circumvents explicit onset detection and instead uses pattern matching of log-lag autocorrelation functions (LLACF) that can be readily extracted from the aforementioned activation functions or similar signal representations. Finally, in Chapter 10, we streamlined the LLACF pattern matching and proposed a novel time–swing representation for tracking micro-rhythmic variations in jazz performances.
We want to conclude this summary with some indications for future work for each of the three main parts of this thesis. These final thoughts and recommendations are of more general nature than the concluding remarks found in the individual chapters.

11.1 Beyond Drum Transcription

As we showed in Chapter 3, the major challenge of state-of-the-art ADT systems usually comes from the interference from other instruments. The superposition of various instruments (e. g., guitar, piano, or singing voice) makes the recognition of certain drum instruments difficult due to the overlaps in both time and frequency (e. g., the KD may overlap with the bass).
As discussed in Section 3.4, we have used two publicly available datasets for our experiments. With respect to their acoustic properties, both corpora feature clean recordings that allow for controlled transcription and source separation experiments. However, real-world music recordings might exhibit different properties that are not reflected well in these datasets. In practice, it is likely that we have to deal with convolutive, time-variant, and non-linear mixtures instead of linear superpositions of single drum sounds. For example, the acoustic conditions of the recording room and the microphone setup may lead to substantial reverberation effects. Furthermore, post-processing like equalization and dynamic compression may be applied to

the drum recordings. Not having these aspects covered in our datasets has two consequences. First, the performance of data-driven methods are likely to deteriorate if the "closed world" of the training data does not match the "open world" of some target data. A typical example is found in speech processing where systems trained with clean speech often fail under noisy or reverberant conditions. Second, any methods involving decompositions based on linear mixture models might be affected when the observed drum mixtures violate their basic assumptions. A possible strategy to cope with the first challenge might be data augmentation [141, 142]. In our case, the amount of training data could be substantially enhanced by applying diverse combinations of audio processing algorithms including reverberation, distortion, and dynamics processing.

As summarized in Table 2.2 (see page 25), many of the existing ADT systems are based on data-driven machine learning approaches. However, with the complexity of music, the difficulty of generating labels, and the restrictions of intellectual property laws, building and sharing annotated datasets becomes a non-trivial task; many of the commonly used datasets are thus limited in different aspects. The most common issue of all the existing drum transcription datasets is the insufficient amount of data. Most of these datasets do not cover a wide range of music genres and playing styles. For instance, RWC-POP [94] only covers Japanese pop music, IDMT-SMT-Drums [37] only covers basic rock and pop music patterns, and ENST-Drums [90] features three different drummers only. This lack of diversity is problematic when ADT systems trained with these datasets are applied to a wider range of music recordings.

A shortcoming shared by most state-of-the-art systems is that the intensity (or loudness) of a drum event is usually ignored in favor of the simple and robust binary representation of the onsets. In principle, activation functions carry some information on onset intensities, but this is usually not encoded in the output of the transcription. Moreover, playing techniques are an important aspect of expressive musical performances. For drum instruments, these techniques include basic rudiments (e.g., roll, paradiddle, drag, and flam) as well as timbral variations (e.g., ghost note, brush, cross stick, and rim shot). In an early attempt to recognize playing techniques, Tindale et al. [203] presented a study on the automatic classification of differences in snare drum timbre induced by striking locations (center, halfway, edge, etc.) and by excitations (strike, rim shot, and brush). Similarly, Prockup et al. [166] tried to classify more expressive gestures on a larger dataset with combinations of different drums, stick heights, stroke intensities, strike positions, and articulations. Souza et al. [191] investigated different playing techniques for cymbals. They differentiated sounds by striking position (bell, body, edge), opening (closed, open, chick), and other special effects such as stopping a cymbal with the playing hand. Although these studies reported promising results for the classification of isolated drum sounds, Wu and Lerch [222] pointed out significant performance gaps for real-world recordings. To obtain a complete transcription in the format of sheet music, more information such as tempo, dynamics, playing styles, or time signatures are required in addition to onset

times. This implies the importance of integrating various MIR systems to the processing chain of ADT systems in order to achieve the ultimate goal of full transcriptions. The research along this direction is still relatively sparse but can be expected to increase as the MIR systems mature.

11.2 Beyond Drum Source Separation

As already mentioned, we have largely relied on generalized Wiener filtering as a means for extracting the desired source signals from a time-frequency representation of the mixture. In particular, in Chapters 4, 5, and 8 we have simply used soft-masks based on estimates of the source magnitude spectrograms as a standard tool without questioning their suitability. In Chapters 6 and 7, we investigated if one could do better. Results indicate that soft-masks based on the STFT magnitude are indeed appropriate for our task of drum sound separation. This is probably due to the broadband nature of the drum sound events that we are dealing with. In contrast to pitched instruments and speech signals, our signals of interest exhibit less sparsity in the time–frequency domain.

We have also seen that soft-masks based on the oracle magnitude spectrograms of the sources do neither yield perfect separation results. This brings us back to think about the additivity assumption [28] underlying many of the separation algorithms. Especially from our experiments with phase reconstruction, it becomes clear that dominant transient sound events such as drum hits will influence large portions of the phase spectrogram, especially during the attack part. Often, we could observe that the corresponding artifacts could be remedied by iterative phase reconstruction, but it could introduce other problems in the decay part. That being said, it would be desirable to cast phase reconstruction as central part of separation algorithm. Besides some mixed results obtained for pitched sounds when using complex NMF [21] or weighted NMF [68], it is largely unclear how to approach that issue.

As a first alternative, we think that drum sound synthesis could be a possible way to go. The idea is to extract expressive parameters from a source signal, which can then be used to control a specialized synthesis algorithm to generate drum sounds. Such strategies have been successfully applied for singing voice synthesis [206].

Second, the recent success of Deep Learning methods for audio waveform generation [63] can be expected to impact source separation applications soon. These have shown considerable qualitative improvements for speech and singing voice generation [17]. Suitably combining them for application to drum sound separation may open a way to overcome the current limitations of conventional source separation based on Wiener filtering.

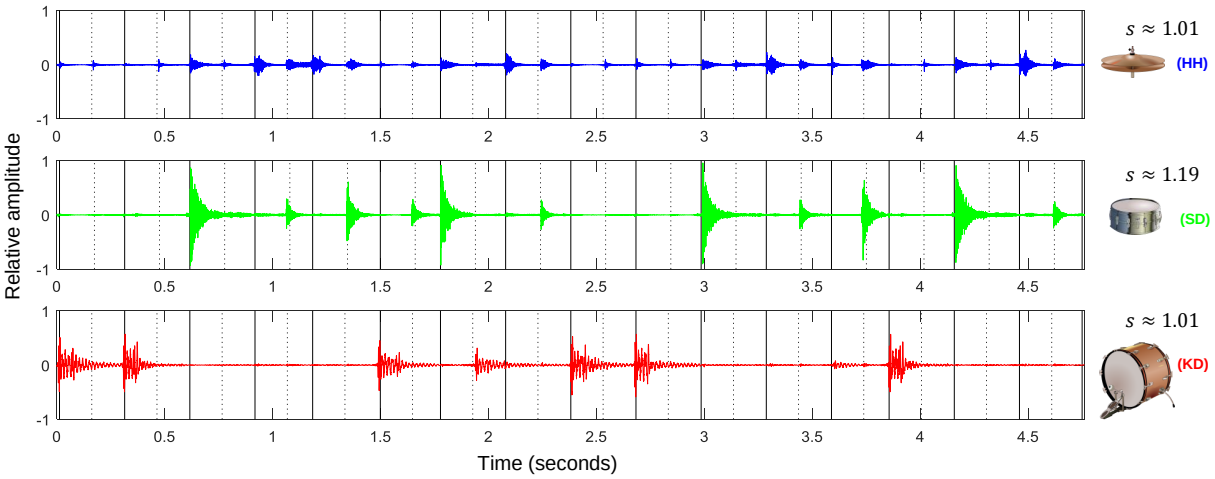

Figure 11.1. Example for sixteenth-note swing ratio estimation in breakbeats.

11.3 Beyond Jazz Analysis

Apart from the relatively well-studied peculiarities of microtiming in jazz music in general and swing ratio estimation in particular (see Chapters 9 and 10), there is a growing interest to investigate related rhythmic phenomena in other styles of music, such as funk and soul. For example, Räsänen et al. [169] conducted a case-study on "I Keep Forgettin"—an emblematic soul recording featuring influential drummer Jeff Porcaro. In the paper, the authors present an in-depth audio analysis of Porcaro's one-handed hi-hat playing technique, revealing characteristic patterns in accents and sixteenth-note inter-onset-intervals. As another example, Davies et al. [35] used rhythmic stimuli that were based on an analysis of the microtiming inherent to the drum solo from "Funky Drummer," originally played by Clyde Stubblefield.

A recently published paper by Frane [80], shows considerable intersections to work presented in this thesis. Interestingly, Frane's subject are breakbeats, the type of drum-only recordings we focused on in Chapter 4. He investigates uneven sixteenth-notes, which can evoke swinging rhythm feel in listeners. Opposed to the swinging eighth-notes in jazz, sixteenth-note swing is often attributed as "Shuffle." By meticulous manual onset annotation in 30 classic breakbeats, Frane shows that the sixteenth-note swing is often used in a much more subtle way than eighth-note swing. He also brings up the notion of "dual" swing ratio for some recordings where he noticed mutual onset deviations in different drum instruments.

To investigate this further, one could combine our breakbeat decomposition method (see Section 4.2) with the swingogram analysis (see Section 10.3). This would open up the possibility to analyze each drum part individually and to circumvent manual onset transcription, instead delivering a more holistic view on each drum instrument's individual swing ratio. Indeed, a promising result can be achieved for one particular example, the "Funky Drummer" breakbeat.

Here, we use the NMFD method as described in Section 4.2 to decompose the recordings in three sources. In this case, since we assume that no manual transcription is available, we revert to audio-informed initialization with generic templates for KD, SD, and HH that have been extracted from isolated drum samples. Then, we use the swingogram extraction method as described in Section 10.3, with the only difference that we virtually double the tempo of the breakbeat so that the swing ratio is estimated on a sixteenth-note grid rather than the usual eighth-note grid.

With the help of the swingogram analysis, we obtain a swing ratio estimate for each of the three drum instruments, as depicted on the right hand side of Figure 11.1. Furthermore, we overlay the separated signal waveforms with the hypothetical eight-note grid (solid gray lines) and sixteenth-note onsets corresponding to estimated swing ratios (dashed gray lines). One can see that the SD exhibits a slightly higher swing ratio than KD and HH. This is also indicated by the attacks portions of the single drum hits that coincide nicely with the dashed sixteenth-note grid. Probably, this contributes to the signature rhythm feel of this classic breakbeat. It remains to be seen, if this phenomenon can be observed in larger datasets as well.

This example serves to further illustrate the potential of applying audio signal processing techniques to cultural artifacts that are of interest to musicologists. Even though it is sometimes not possible to formalize the respective research questions as a computational task, there are many examples where (semi-)automatic music analysis with MIR methods can already bring a considerable benefit for researchers in other domains. For example, Pfleiderer et al. [164] feature several case studies that begin with "close readings", i. e., in-depth manual analysis of a particular music recording by an expert. These are then extended by "distant readings", i. e., statistical evaluation of larger music corpora. These serve to test the hypotheses formulated by the musicologists in a more data-rich manner, similar to our work in Section 10.5.2. This kind of interdisciplinary cooperation and mutual inspiration between engineering and musicology has been an important driver during my work on this thesis.

Bibliography

[1] Mike Adamo. *The Breakbeat Bible*. Hudson Music, Briarcliff, NY, USA, 2010. ISBN 978-1423496335.

[2] David S. Alves, Jouni Paulus, and José Fonseca. Drum transcription from multichannel recordings with non-negative matrix factorization. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, pages 894–898, Glasgow, Scotland, UK, August 2009.

[3] Stefan Balke, Jakob Abeßer, Jonathan Driedger, Christian Dittmar, and Meinard Müller. Towards evaluating multiple predominant melody annotations in jazz recordings. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 246–252, New York City, USA, August 2016.

[4] Stefan Balke, Christian Dittmar, Jakob Abeßer, and Meinard Müller. Data-driven solo voice enhancement for jazz music retrieval. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 196–200, New Orleans, Louisiana, USA, March 2017.

[5] Stefan Balke, Christian Dittmar, and Meinard Müller. Ansätze zur datengetriebenen Transkription einstimmiger Jazzsoli. In *Proceedings of the Deutsche Jahrestagung für Akustik (DAGA)*, pages 1530–1532, München, Germany, March 2018.

[6] Mark A. Bartsch and Gregory H. Wakefield. Audio thumbnailing of popular music using chroma-based representations. *IEEE Transactions on Multimedia*, 7(1):96–104, 2005.

[7] Eric Battenberg. *Techniques for Machine Understanding of Live Drum Performances*. PhD thesis, University of California at Berkeley, 2012.

[8] Eric Battenberg, Victor Huang, and David Wessel. Live drum separation using probabilistic spectral clustering based on the Itakura-Saito divergence. In *Proceedings of the Audio Engineering Society (AES) Conference on Time-Frequency Processing in Audio*, Helsinki, Finland, March 2012.

[9] Juan Pablo Bello, Emmanuel Ravelli, and Mark B. Sandler. Drum sound analysis for the manipulation of rhythm in drum loops. In *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, Toulouse, France, May 2006.

[10] Fernando Benadon. Slicing the beat: Jazz eighth-notes as expressive microrhythm. *Ethnomusicology*, 50(1):73–98, 2006.

[11] Laurent Benaroya, Lorcan McDonagh, Frédéric Bimbot, and Rémi Gribonval. Non negative sparse representation for Wiener based source separation with a single sensor. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 613–616, Hong Kong, China, April 2003.

[12] Emmanouil Benetos and Simon Dixon. Multiple-instrument polyphonic music transcription using a temporally constrained shift-invariant model. *The Journal of the Acoustical Society of America (JASA)*, 133(3):1727–1741, 2013.

[13] Emmanouil Benetos, Simon Dixon, Dimitrios Giannoulis, Holger Kirchhoff, and Anssi Klapuri. Automatic music transcription: challenges and future directions. *Journal of Intelligent Information Systems*, 41(3):407–434, 2013.

[14] Emmanouil Benetos, Roland Badeau, Tillman Weyde, and Gaël Richard. Template adaptation for improving automatic music transcription. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 175–180, Taipei, Taiwan, October 2014.

[15] Emmanouil Benetos, Sebastian Ewert, and Tillman Weyde. Automatic transcription of pitched and unpitched sounds from polyphonic music. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3107–3111, Florence, Italy, May 2014.

[16] Paul F. Berliner. *Thinking in Jazz. The Infinite Art of Improvisation.* University of Chicago Press, 1994.

[17] Merlijn Blaauw and Jordi Bonada. A neural parametric singing synthesizer. *CoRR*, abs/1704.03809, 2017.

[18] Sebastian Böck and Markus Schedl. Polyphonic piano note transcription with recurrent neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 121–124, Kyoto, Japan, March 2012.

[19] Sebastian Böck, Andreas Arzt, Florian Krebs, and Markus Schedl. Online real-time onset detection with recurrent neural networks. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, York, UK, September 2012.

[20] Albert S. Bregman. *Auditory Scene Analysis: The Perceptual Organization of Sound.* MIT Press, 1990.

[21] James Bronson and Philippe Depalle. Phase constrained complex NMF: Separating overlapping partials in mixtures of harmonic musical sources. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7475–7479, Florence, Italy, May 2014.

[22] Walter Gerard Busse. Toward objective measurement and evaluation of jazz piano performance via midi-based groove quantize templates. *Music Perception*, 19(3):443–461, 2002.

[23] Matthew W. Butterfield. Why do jazz musicians swing their eighth notes? *Music Theory Spectrum*, 33(1):3–26, 2011.

[24] Francisco J. Cañadas-Quesada, Pedro Vera-Candeas, Nicolás Ruiz-Reyes, Julio J. Carabias-Orti, and Pablo Cabañas Molero. Percussive/harmonic sound separation by non-negative matrix factorization with smoothness/sparseness constraints. *EURASIP Journal on Audio, Speech and Music Processing*, 26, 2014.

[25] Francisco J. Cañadas-Quesada, Derry FitzGerald, Pedro Vera-Candeas, and Nicolás Ruiz-Reyes. Harmonic-percussive sound separation using rhythmic information from non-negative matrix factorization in single-channel music recordings. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 276–282, Edinburgh, UK, September 2017.

[26] Chris Cannam, Christian Landone, and Mark B. Sandler. Sonic visualiser: An open source application for viewing, analysing, and annotating music audio files. In *Proceedings of the International Conference on Multimedia*, pages 1467–1468, Florence, Italy, October 2010.

[27] Estefanía Cano, Jakob Abeßer, Christian Dittmar, and Gerald Schuller. Influence of phase, magnitude and location of harmonic components in the perceived quality of extracted solo signals. In *Proceedings of the Audio Engineering Society (AES) Conference on Semantic Audio*, pages 247–252, Ilmenau, Germany, July 2011.

[28] Estefanía Cano, Christian Dittmar, and Gerald Schuller. Re-thinking sound separation: Prior information and additivity constraint in separaton algorithms. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, Maynooth, Ireland, September 2013.

[29] Estefanía Cano, Mark Plumbley, and Christian Dittmar. Phase-based harmonic percussive separation. In *Proceedings of the Annual Conference of the International Speech Communication Association (Interspeech)*, pages 1628–1632, Singapore, September 2014.

[30] Estefanía Cano, Christian Dittmar, Jakob Abeßer, Christian Kehling, and Sascha Grollmisch. Music technology and education. In Rolf Bader, editor, *Springer Handbook on Systematic Musicology*, pages 855–871. Springer, Berlin, Heidelberg, 2018. ISBN 978-3-662-55002-1.

[31] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, , Dzmitry Bahdanau, Fehti Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, October 2014.

[32] Geoffrey L. Collier and James Lincoln Collier. A study of timing in two louis armstrong solos. *Music Perception*, 19(3):463–483, 2002.

[33] Umut Şimşekli and Ali Taylan Cemgil. Score guided musical source separation using generalized coupled tensor factorization. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, pages 2639–2643, Bucharest, Romania, August 2012.

[34] Umut Şimşekli, Antti Jylhä, Cumhur Erkut, and Ali Taylan Cemgil. Real-time recognition of percussive sounds by a model-based method. *EURASIP Journal on Audio, Speech, and Music Processing*, 2011, 2011.

[35] Matthew E. P. Davies, Guy Madison, Pedro Silva, and Fabien Gouyon. The effect of microtiming deviations on the perception of groove in short rhythms. *Music Perception*, 30(5):497–510, 2013.

[36] Sven Degroeve, Koen Tanghe, Bernard De Baets, Marc Leman, and Jean-Pierre Martens. A simulated annealing optimization of audio features for drum classification. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 482–487, London, UK, September 2005.

[37] Christian Dittmar and Daniel Gärtner. Real-time transcription and separation of drum recordings based on NMF decomposition. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 187–194, Erlangen, Germany, September 2014.

[38] Christian Dittmar and Meinard Müller. Towards transient restoration in score-informed audio decomposition. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 145–152, Trondheim, Norway, December 2015.

[39] Christian Dittmar and Meinard Müller. Reverse engineering the Amen break – score-informed separation and restoration applied to drum recordings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(9):1531–1543, 2016.

[40] Christian Dittmar and Christian Uhle. Further steps towards drum transcription of polyphonic music. In *Proceedings of the Audio Engineering Society Convention (AES)*, Berlin, Germany, May 2004.

[41] Christian Dittmar, Kay F. Hildebrand, Daniel Gärtner, Manuel Winges, Florian Müller, and Patrick Aichroth. Audio forensics meets music information retrieval – a toolbox for inspection of music plagiarism. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, pages 1249–1253, Bucharest, Romania, August 2012.

[42] Christian Dittmar, Jonathan Driedger, and Meinard Müller. A separate and restore approach to score-informed music decomposition. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, October 2015.

[43] Christian Dittmar, Bernhard Lehner, Thomas Prätzlich, Meinard Müller, and Gerhard Widmer. Cross-version singing voice detection in classical opera recordings. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 618–624, Málaga, Spain, October 2015.

[44] Christian Dittmar, Martin Pfleiderer, and Meinard Müller. Automated estimation of ride cymbal swing ratios in jazz recordings. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 271–277, Málaga, Spain, October 2015.

[45] Christian Dittmar, Thomas Prätzlich, and Meinard Müller. Towards cross-version singing voice detection. In *Proceedings of the Deutsche Jahrestagung für Akustik (DAGA)*, pages 1503–1506, Nürnberg, Germany, March 2015.

[46] Christian Dittmar, Jonathan Driedger, Meinard Müller, and Jouni Paulus. An experimental approach to generalized Wiener filtering in music source separation. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Budapest, Hungary, August 2016.

[47] Christian Dittmar, Patricio López-Serrano, and Meinard Müller. Unifying local and global methods for harmonic-percussive source separation. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Calgary, Canada, April 2018.

[48] Christian Dittmar, Martin Pfleiderer, Stefan Balke, and Meinard Müller. A swingogram representation for tracking micro-rhythmic variation in jazz performances. *Journal of New Music Research*, 47 (2):97–113, 2018.

[49] Simon Dixon. Onset detection revisited. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 133–137, Montreal, Quebec, Canada, September 2006.

[50] Simon Dixon, Fabien Gouyon, and Gerhard Widmer. Towards characterisation of music via rhythmic patterns. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Barcelona, Spain, October 2004.

[51] Jonathan Driedger, Harald Grohganz, Thomas Prätzlich, Sebastian Ewert, and Meinard Müller. Score-informed audio decomposition and applications. In *Proceedings of the ACM International Conference on Multimedia (ACM-MM)*, pages 541–544, Barcelona, Spain, 2013.

[52] Jonathan Driedger, Meinard Müller, and Sascha Disch. Extending harmonic-percussive separation of audio signals. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 611–616, Taipei, Taiwan, October 2014.

[53] Jonathan Driedger, Meinard Müller, and Sebastian Ewert. Improving time-scale modification of music signals using harmonic-percussive separation. *IEEE Signal Processing Letters*, 21(1):105–109, 2014.

[54] Jonathan Driedger, Thomas Prätzlich, and Meinard Müller. Let It Bee – Towards NMF-inspired audio mosaicing. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 350–356, Málaga, Spain, 2015.

[55] John C. Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

[56] Jean-Louis Durrieu, Bertrand David, and Gaël Richard. A musically motivated mid-level representation for pitch estimation and musical audio source separation. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1180–1191, 2011.

[57] Chris Duxbury, Mike Davies, and Mark B. Sandler. Improved time-scaling of musical audio using phase locking at transients. In *Proceedings of the Audio Engineering Society (AES) Convention*, Munich, Germany, May 2002. Preprint 5530.

[58] Georgi Dzhambazov. Towards a drum transcription system aware of bar position. In *Proceedings of the Audio Engineering Society Conference on Semantic Audio (AES)*, London, UK, January 2014.

[59] Bernd Edler. Codierung von Audiosignalen mit überlappender Transformation und adaptiven Fensterfunktionen. *Frequenz*, 43(9):252–256, September 1989.

[60] Daniel P.W. Ellis. Beat tracking by dynamic programming. *Journal of New Music Research*, 36(1): 51–60, 2007.

[61] Mark C. Ellis. An analysis of "swing" subdivision and asynchronization in three jazz saxophonists. *Perceptual and Motor Skills*, 73(3):707–713, 1991.

[62] Valentin Emiya, Emmanuel Vincent, Niklas Harlander, and Volker Hohmann. Subjective and objective quality assessment of audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2046–2057, 2011.

[63] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. Neural audio synthesis of musical notes with wavenet autoencoders. In *Proceedings of the International Conference on Machine Learning ICML*, pages 1068–1077, Sydney, Australia, August 2017.

[64] Arndt Eppler, Andreas Männchen, Jakob Abeßer, Christof Weiß, and Klaus Frieler. Automatic style classification of jazz records with respect to rhythm, tempo, and tonality. In *Proceedings of the Conference on Interdisciplinary Musicology (CIM)*, pages 162–167, December 2014.

[65] Antti J. Eronen. Musical instrument recognition using ICA-based transform of features and discriminatively trained HMMs. In *Proceedings of the International Symposium on Signal Processing and Its Applications (ISSPA)*, volume 2, pages 133–136, Paris, France, July 2003.

[66] Sebastian Ewert and Meinard Müller. Using score-informed constraints for NMF-based source separation. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 129–132, Kyoto, Japan, March 2012.

[67] Sebastian Ewert, Bryan Pardo, Meinard Müller, and Mark Plumbley. Score-informed source separation for musical audio recordings: An overview. *IEEE Signal Processing Magazine*, 31(3): 116–124, April 2014.

[68] Sebastian Ewert, Mark D. Plumbley, and Mark B. Sandler. Accounting for phase cancellations in non-negative matrix factorization using weighted distances. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 649–653, Florence, Italy, May 2014.

[69] Florian Eyben, Sebastian Böck, Björn Schuller, and Alex Graves. Universal onset detection with bidirectional long short-term memory neural networks. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 589–594, Utrecht, The Netherlands, August 2010.

[70] Derry FitzGerald. *Automatic drum transcription and source separation.* PhD thesis, Dublin Institute of Technology, Dublin, Ireland, 2004.

[71] Derry FitzGerald. Harmonic/percussive separation using median filtering. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 246–253, Graz, Austria, September 2010.

[72] Derry FitzGerald and Rajesh Jaiswahl. On the use of masking filters in sound source separation. In *Proceedings of the International Confenference on Digital Audio Effects (DAFx)*, York, UK, September 2012.

[73] Derry FitzGerald and Jouni Paulus. Unpitched percussion transcription. In Anssi Klapuri and Manuel Davy, editors, *Signal Processing Methods for Music Transcription*, pages 131–162. Springer US, Boston, MA, USA, 2006.

[74] Derry FitzGerald, Bob Lawlor, and Eugene Coyle. Sub-band independent subspace analysis for drum transcription. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 65–69, Hamburg, Germany, September 2002.

[75] Derry FitzGerald, Bob Lawlor, and Eugene Coyle. Drum transcription in the presence of pitched instruments using prior subspace analysis. In *Proceedings of the Irish Signals and Systems Conference (ISSC)*, Limerick, Ireland, July 2003.

[76] Derry FitzGerald, Bob Lawlor, and Eugene Coyle. Prior subspace analysis for drum transcription. In *Proceedings of the Audio Engineering Society (AES) Convention*, Amsterdam, The Netherlands, March 2003.

[77] Derry FitzGerald, Eugene Coyle, and Matt Cranitch. Using tensor factorisation models to separate drums from polyphonic music. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, Camo, Italy, September 2009.

[78] Derry FitzGerald, Antoine Liutkus, Zafar Rafii, Bryan Pardo, and Laurent Daudet. Harmonic/percussive separation using Kernel Additive Modelling. In *Irish Signals and Systems Conference (IET)*, pages 35–40, Limerick, Ireland, 2014.

[79] Jonathan Foote and Shingo Uchihashi. The beat spectrum: A new approach to rhythm analysis. In *Proceedings of the International Conference on Multimedia and Expo (ICME)*, pages 881–884, Tokyo, Japan, August 2001.

[80] Andrew V. Frane. Swing rhythm in classic drum breaks from hip-hop's breakbeat canon. *Music Perception*, 34(3):291–302, 2017.

[81] Anders Friberg and Andreas Sundström. Swing ratios and ensemble timing in jazz performance: Evidence for a common rhythmic pattern. *Music Perception*, 19(3):333–349, 2002.

[82] Klaus Frieler, Wolf-Georg Zaddach, Jakob Abeßer, and Martin Pfleiderer. Introducing the Jazzomat project and the melospy library. In *Third International Workshop on Folk Music Analysis*, 2013.

[83] Joachim Fritsch and Mark D. Plumbley. Score informed audio source separation using constrained nonnegative matrix factorization and score synthesis. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 888–891, Vancouver, Canada, May 2013.

[84] Richard Füg, Andreas Niedermeier, Jonathan Driedger, Sascha Disch, and Meinard Müller. Harmonic-percussive-residual sound separation using the structure tensor on spectrograms. In *Proceedings of*

the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pages 445–449, Shanghai, China, March 2016.

[85] Nicolai Gajhede, Oliver Beck, and Hendrik Purwins. Convolutional neural networks with batch normalization for classifying hi-hat, snare, and bass percussion sound samples. In *Proceedings of the Audio Mostly: A Conference on Interaction with Sound*, pages 111–115, Norrköping, Sweden, October 2016.

[86] Timo Gerkmann, Martin Krawczyk-Becker, and Jonathan Le Roux. Phase processing for single-channel speech enhancement: History and recent advances. *IEEE Signal Processing Magazine*, 32 (2):55–66, March 2015. ISSN 1053-5888.

[87] Olivier Gillet and Gaël Richard. Automatic transcription of drum loops. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 269–272, Montreal, Quebec, Canada, May 2004.

[88] Olivier Gillet and Gaël Richard. Drum track transcription of polyphonic music signals using noise subspace projection. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 92–99, London, UK, September 2005.

[89] Olivier Gillet and Gaël Richard. Automatic transcription of drum sequences using audiovisual features. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 205–208, Philadelphia, Pennsylvania, USA, March 2005.

[90] Olivier Gillet and Gaël Richard. ENST-drums: an extensive audio-visual database for drum signals processing. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 156–159, Victoria, Canada, October 2006.

[91] Olivier Gillet and Gaël Richard. Supervised and unsupervised sequence modelling for drum transcription. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 219–224, Vienna, Austria, September 2007.

[92] Olivier Gillet and Gaël Richard. Transcription and separation of drum signals from polyphonic music. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(3):529–540, 2008.

[93] Volker Gnann and Martin Spiertz. Inversion of short-time Fourier transform magnitude spectrograms with adaptive window lengths. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP)*, pages 325–328, Taipei, Taiwan, April 2009.

[94] Masataka Goto, Hiroki Hashiguchi, Takuichi Nishimura, and Ryuichi Oka. RWC music database: Music genre database and musical instrument sound database. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 229–230, Baltimore, Maryland, USA, October 2003.

[95] Fabien Gouyon, François Pachet, and Olivier Delerue. On the use of zero-crossing rate for an application of classification of percussive sounds. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, Verona, Italy, December 2000.

[96] Daniel W. Griffin and Jae S. Lim. Signal estimation from modified short-time Fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(2):236–243, 1984.

[97] Peter Grosche and Meinard Müller. Extracting predominant local pulse information from music recordings. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(6):1688–1701, 2011.

[98] Peter Grosche and Meinard Müller. Tempogram Toolbox: MATLAB tempo and pulse analysis of music recordings. In *Late-Breaking and Demo Session of the Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Miami, Florida, USA, October 2011.

[99] Peter Grosche, Meinard Müller, and Frank Kurth. Cyclic tempogram – a mid-level tempo representation for music signals. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5522–5525, Dallas, Texas, USA, March 2010.

[100] Matthias Gruhne and Christian Dittmar. Improving rhythmic pattern features based on logarithmic preprocessing. In *Proceedings of the Audio Engineering Society (AES) Convention*, Munich, Germany, May 2009.

[101] Amaury Hazan. Towards automatic transcription of expressive oral percussive performances. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI)*, pages 296–298, San Diego, California, USA, January 2005.

[102] Marko Leonard Helén and Tuomas Virtanen. Separation of drums from polyphonic music using nonnegative matrix factorization and support vector machine. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Antalya, Turkey, September 2005.

[103] Romain Hennequin, Roland Badeau, and Bertrand David. NMF with time–frequency activations to model nonstationary audio events. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):744–753, 2011.

[104] Jürgen Herre and James D. Johnston. Enhancing the performance of perceptual audio coders by using temporal noise shaping (TNS). In *Proceedings of the Audio Engineering Society (AES) Convention*, Los Angeles, California, USA, November 1996.

[105] Perfecto Herrera, Alexandre Yeterian, and Fabien Gouyon. Automatic classification of drum sounds: A comparison of feature selection methods and classification techniques. In *Proceedings of the International Conference on Music and Artificial Intelligence (ICMAI)*, pages 69–80, Edinburgh, Scotland, UK, September 2002.

[106] Perfecto Herrera, Amaury Dehamel, and Fabien Gouyon. Automatic labeling of unpitched percussion sounds. In *Proceedings of the Audio Engineering Society Convention (AES)*, Amsterdam, The Netherlands, March 2003.

[107] Perfecto Herrera, Vegard Sandvold, and Fabien Gouyon. Percussion-related semantic descriptors of music audio files. In *Audio Engineering Society Conference: Metadata for Audio (AES)*, pages 69–73, London, UK, June 2004.

[108] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

[109] Jason A. Hockman. *An ethnographic and technological study of breakbeats in Hardcore, Jungle, and Drum & Bass*. PhD thesis, McGill University, Montreal, Quebec, Canada, 2012.

[110] Jason A. Hockman, Matthew E. P. Davies, and Ichiro Fujinaga. Computational strategies for breakbeat classification and resequencing in Hardcore, Jungle and Drum & Bass. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 337–342, Trondheim, Norway, December 2015.

[111] André Holzapfel and Yannis Stylianou. A scale transform based method for rhythmic similarity of music. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 317–320, Taipei, Taiwan, April 2009.

[112] Andre Holzapfel and Yannis Stylianou. Scale transform in rhythmic similarity of music. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(1):176–185, 2011.

[113] Henkjan Honing and W. Bas de Haas. Swing once more: Relating timing and tempo in expert jazz drumming. *Music Perception*, 25(5):471–476, 2008.

[114] Jesper Højvang Jensen, Mads Græsbøll Christensen, and Søren Holdt Jensen. A tempo-insensitive representation of rhythmic patterns. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, pages 1509–1512, Glasgow, Scotland, August 2009.

[115] Kristoffer Jensen. Multiple scale music segmentation using rhythm, timbre, and harmony. *EURASIP Journal on Advances in Signal Processing*, 2007.

[116] Maximos A. Kaliakatsos-Papakostas, Andreas Floros, Michael N. Vrahatis, and Nikolaos Kanellopoulos. Real-time drums transcription with characteristic bandpass filtering. In *Proceedings of the Audio Mostly: A Conference on Interaction with Sound*, pages 152–159, Corfu, Greece, September 2012.

[117] Hirokazu Kameoka, Nobutaka Ono, Kunio Kashino, and Shigeki Sagayama. Complex NMF: A new sparse representation for acoustic signals. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 3437–3440, Taipei, Taiwan, April 2009.

[118] Franz Kerschbaumer. *Miles Davis: Stilkritische Untersuchungen zur musikalischen Entwicklung seines Personalstils*. Studies in jazz research. Akademische Druck und Verlagsanstalt, 1978.

[119] Minje Kim, Jiho Yoo, Kyeongok Kang, and Seungjin Choi. Nonnegative matrix partial co-factorization for spectral and temporal drum source separation. *IEEE Journal of Selected Topics Signal Processing*, 5(6):1192–1204, 2011.

[120] Brian King, Cédric Févotte, and Paris Smaragdis. Optimal cost function and magnitude power for NMF-based speech separation and music interpolation. In *Proceedings of the IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, Santander, Spain, September 2012.

[121] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[122] Elias Kokkinis, Alexandros Tsilfidis, Thanos Kostis, and Kostas Karamitas. A new DSP tool for drum leakage suppression. In *Proceedings of the Audio Engineering Society Convention (AES)*, New York City, USA, October 2013.

[123] Frank Kurth. The shift-ACF: Detecting multiply repeated signal components. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, 2013.

[124] Frank Kurth, Thorsten Gehrmann, and Meinard Müller. The cyclic beat spectrum: Tempo-related audio features for time-scale invariant audio identification. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 35–40, Victoria, Canada, October 2006.

[125] Clément Laroche, Hélène Papadopoulos, Matthieu Kowalski, and Gaël Richard. Drum extraction in single channel audio signals using multi-layer non negative matrix factor deconvolution. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 46–50, New Orleans, Louisiana, USA, March 2017.

[126] Jonathan Le Roux, Nobutaka Ono, and Shigeki Sagayama. Explicit consistency constraints for STFT spectrograms and their application to phase reconstruction. In *Proceedings of the ISCA Tutorial and Research Workshop on Statistical And Perceptual Audition (SAPA)*, pages 23–28, Brisbane, Australia, September 2008.

[127] Jonathan Le Roux, Hirokazu Kameoka, Nobutaka Ono, and Shigeki Sagayama. Fast signal reconstruction from magnitude STFT spectrogram based on spectrogram consistency. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 397–403, Graz, Austria, September 2010.

[128] Jonathan Le Roux, Hirokazu Kameoka, Nobutaka Ono, and Shigeki Sagayama. Phase initialization schemes for faster spectrogram-consistency-based signal reconstruction. In *Proceedings of the Acoustical Society of Japan Autumn Meeting*, pages 601–602, September 2010.

[129] Jonathan Le Roux, Emmanuel Vincent, Yuu Mizuno, Hirokazu Kameoka, Nobutaka Ono, and Shigeki Sagayama. Consistent Wiener filtering: Generalized time-frequency masking respecting spectrogram consistency. In *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA)(Lecture Notes in Computer Science, Vol. 6365)*, pages 89–96, St. Malo, France, September 2010.

[130] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Proceedings of the Neural Information Processing Systems (NIPS)*, pages 556–562, Denver, Colorado, USA, November 2000.

[131] Matthias Leimeister, Daniel Gärtner, and Christian Dittmar. Rhythmic classification of electronic dance music. In *Proceedings of the Audio Engineering Society Conference on Semantic Audio (AES)*, London, UK, January 2014.

[132] Yipeng Li, John Woodruff, and DeLiang Wang. Monaural musical sound separation based on pitch and common amplitude modulation. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(7):1361–1371, 2009.

[133] Henry Lindsay-Smith, Skot McDonald, and Mark B. Sandler. Drumkit transcription via convolutive NMF. In *Proceedings of the International Conference on Digital Audio Effects Conference (DAFx)*, York, UK, September 2012.

[134] Antoine Liutkus and Roland Badeau. Generalized Wiener filtering with fractional power spectrograms. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 266–270, Brisbane, Australia, April 2015.

[135] Antoine Liutkus, Derry FitzGerald, Zafar Rafii, Bryan Pardo, and Laurent Daudet. Kernel additive models for source separation. *IEEE Transactions on Signal Processing*, 62(16):4298–4310, 2014.

[136] Patricio López-Serrano, Christian Dittmar, Jonathan Driedger, and Meinard Müller. Towards modeling and decomposing loop-based electronic music. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 502–508, New York City, USA, August 2016.

[137] Patricio López-Serrano, Christian Dittmar, and Meinard Müller. Mid-level audio features based on cascaded harmonic-residual-percussive separation. In *Proceedings of the Audio Engineering Society Conference on Semantic Audio (AES)*, pages 32–44, Erlangen, Germany, June 2017.

[138] Patricio López-Serrano, Christian Dittmar, and Meinard Müller. Finding drum breaks in digital music recordings. In *Proceedings of the International Symposium on Computer Music Multidisciplinary Research (CMMR)*, pages 68–79, Porto, Portugal, September 2017.

[139] Ugo Marchand and Geoffroy Peeters. The modulation scale spectrum and its application to rhythm-content description. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 167–172, Erlangen, Germany, September 2014.

[140] Ugo Marchand and Geoffroy Peeters. Swing ratio estimation. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 423–428, Trondheim, Norway, December 2015.

[141] Matthias Mauch and Sebastian Ewert. The audio degradation toolbox and its application to robustness evaluation. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 83–88, Curitiba, Brazil, November 2013.

[142] Brian McFee, Eric J. Humphrey, and Juan Pablo Bello. A software framework for musical data augmentation. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 248–254, Málaga, Spain, October 2015.

[143] Marius Miron, Matthew E. P. Davies, and Fabien Gouyon. An open-source drum transcription system for Pure Data and Max/MSP. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 221–225, Vancouver, BC, Canada, May 2013.

[144] Marius Miron, Matthew E. P. Davies, and Fabien Gouyon. Improving the real-time performance of a causal audio drum transcription system. In *Proceedings of the Sound and Music Computing Conference (SMC)*, pages 402–407, Stockholm, Sweden, July 2013.

[145] Arnaud Moreau and Arthur Flexer. Drum transcription in polyphonic music using non-negative matrix factorisation. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 353–354, Vienna, Austria, September 2007.

[146] Meinard Müller. *Fundamentals of Music Processing*. Springer Verlag, 2015. ISBN 978-3-319-21944-8.

[147] Meinard Müller, Thomas Prätzlich, and Christian Dittmar. Freischütz Digital – When computer science meets musicology. In Kristina Richts and Peter Stadler, editors, *Festschrift für Joachim Veit zum 60. Geburtstag*, pages 551–573, München, Germany, 2016. Allitera.

[148] Tomohiko Nakamura and Hiokazu Kameoka. Fast signal reconstruction from magnitude spectrogram of continuous wavelet transform based on spectrogram consistency. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 129–135, Erlangen, Germany, September 2014.

[149] Tomoyasu Nakano, Jun Ogata, Masataka Goto, and Yuzuru Hiraga. A drum pattern retrieval method by voice percussion. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 550–553, Barcelona, Spain, October 2004.

[150] Yurii Nesterov. A method of solving a convex programming problem with convergence rate o(1/sqr(k)). *Soviet Mathematics Doklady*, 27(2):372–376, 1983.

[151] Oliver Niemeyer and Bernd Edler. Detection and extraction of transients for audio coding. In *Proceedings of the Audio Engineering Society (AES) Convention*, Paris, France, May 2006.

[152] Nobutaka Ono, Kenichi Miyamoto, Jonathan LeRoux, Hirokazu Kameoka, and Shigeki Sagayama. Separation of a monaural audio signal into harmonic/percussive components by complementary diffusion on spectrogram. In *European Signal Processing Conference (EUSIPCO)*, pages 240–244, Lausanne, Switzerland, 2008.

[153] Alexey Ozerov, Cédric Févotte, Raphaël Blouet, and Jean-Louis Durrieu. Multichannel nonnegative tensor factorization with structured constraints for user-guided audio source separation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 257–260, Prague, Czech Republic, 2011.

[154] Elias Pampalk, Perfecto Herrera, and Masataka Goto. Computational models of similarity for drum samples. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2):408–423, 2008.

[155] Jeongsoo Park and Kyogu Lee. Harmonic-percussive source separation using harmonicity and sparsity constraints. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 148–154, Málaga, Spain, October 2015.

[156] Jeongsoo Park, Jaeyoung Shin, and Kyogu Lee. Exploiting continuity/discontinuity of basis vectors in spectrogram decomposition for harmonic-percussive sound separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(5):1061–1074, May 2017.

[157] Will Parsons and Ernest Cholakis. It don't mean a thing if it ain't dang, dang-a dang! *Downbeat*, 52(8):61, 1995.

[158] Jouni Paulus. *Signal Processing Methods for Drum Transcription and Music Structure Analysis*. PhD thesis, Tampere University of Technology, Tampere, Finland, 2009.

[159] Jouni Paulus and Anssi Klapuri. Drum sound detection in polyphonic music with hidden markov models. *EURASIP Journal on Audio, Speech, and Music Processing*, 2009(14), 2009.

[160] Jouni Paulus and Tuomas Virtanen. Drum transcription with non-negative spectrogram factorisation. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Antalya, Turkey, September 2005.

[161] Geoffroy Peeters. Rhythm classification using spectral rhythm patterns. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 644–647, London, UK, September 2005.

[162] Nathanaël Perraudin, Peter Balazs, and Peter L. Søndergaard. A fast Griffin-Lim algorithm. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, October 2013.

[163] Martin Pfleiderer. *Rhythmus: Psychologische, theoretische und stilanalytische Aspekte populärer Musik*. Transcript, 2006.

[164] Martin Pfleiderer, Klaus Frieler, Jakob Jakob Abeßer, Wolf-Georg Zaddach, and Benjamin Burkhart. *Inside the Jazzomat*. Schott Campus, Mainz, Germany, 2017. ISBN 978-3-95983-124-6.

[165] Thomas Prätzlich, Rachel Bittner, Antoine Liutkus, and Meinard Müller. Kernel additive modeling for interference reduction in multi-channel music recordings. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 584–588, Brisbane, Australia, April 2015.

[166] Matthew Prockup, Erik M. Schmidt, Jeffrey Scott, and Youngmoo E. Kim. Toward understanding expressive percussion through content based analysis. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 143–148, Curitiba, Brazil, November 2013.

[167] Matthew Prockup, Andreas F. Ehmann, Fabien Gouyon, Erik M. Schmidt, and Youngmoo E. Kim. Modeling musical rhythm at scale with the music genome project. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, October 2015.

[168] Stanislaw Andrzej Raczynski, Nobutaka Ono, and Shigeki Sagayama. Multipitch analysis with harmonic nonnegative matrix approximation. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 381–386, Vienna, Austria, September 2007.

[169] Esa Räsänen, Otto Pulkkinen, Tuomas Virtanen, Manfred Zollner, and Holger Hennig. Fluctuations of hi-hat timing and dynamics in a virtuoso drum track of a popular music recording. *PLOS ONE*, 10(6):1–16, June 2015.

[170] Peter Reinholdsson. Approaching jazz performances empirically. some reflections on methods and problems. *Action and perception in rhythm and music*, 55:105–125, 1987.

[171] Olivier Rioul and Martin Vetterli. Wavelets and signal processing. *IEEE Signal Processing Magazine*, 8(4):14–38, October 1991.

[172] Axel Röbel. A new approach to transient processing in the phase vocoder. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 344–349, London, UK, September 2003.

[173] Axel Röbel and Xavier Rodet. Efficient spectral envelope estimation and its application to pitch shifting and envelope preservation. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 344–349, Madrid, Spain, September 2005.

[174] Axel Röbel, Jordi Pons, Marco Liuni, and Mathieu Lagrange. On automatic drum transcription using non-negative matrix deconvolution and Itakura Saito divergence. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 414–418, Brisbane, Australia, April 2015.

[175] Richard Franklin Rose. *An Analysis of Timing in Jazz Rhythm Section Performances*. PhD thesis, University of Texas, 1989.

[176] Mathias Rossignol, Mathieu Lagrange, Grégoire Lafay, and Emmanouil Benetos. Alternate level clustering for drum transcription. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, pages 2023–2027, Nice, France, August 2015.

[177] Pierre Roy, François Pachet, and Sergio Krakowski. Improving the classification of percussive sounds with analytical features: A case study. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 229–232, Vienna, Austria, September 2007.

[178] Justin Salamon, Emilia Gómez, Daniel P. W. Ellis, and Gaël Richard. Melody extraction from polyphonic music signals: Approaches, applications, and challenges. *IEEE Signal Processing Magazine*, 31(2):118–134, 2014.

[179] Vegard Sandvold, Fabien Gouyon, and Perfecto Herrera. Percussion classification in polyphonic audio recordings using localized sound models. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 537–540, Barcelona, Spain, October 2004.

[180] W. Andrew Schloss. *On the Automatic Transcription of Percussive Music - From Acoustic Signal to High-Level Analysis*. PhD thesis, Stanford University, 1985.

[181] Simon Scholler and Hendrik Purwins. Sparse coding for drum sound classification and its use as a similarity measure. In *Proceedings of the International Workshop on Machine Learning and Music (MML)*, pages 9–12, Florence, Italy, October 2010.

[182] Simon Scholler and Hendrik Purwins. Sparse approximations for drum sound classification. *Journal of Selected Topics Signal Processing*, 5(5):933–940, 2011.

[183] Xavier Serra. Musical sound modeling with sinusoids plus noise. In C. Roads, S. Pope, A. Picialli, and G. De Poli, editors, *Musical Signal Processing*. Swets & Zeitlinger, 1997.

[184] Madhusudana Shashanka. *Latent Variable Framework for Modeling and Separating Single Channel Acoustic Sources*. PhD thesis, Department of Cognitive and Neural Systems, Boston University, 2007.

[185] Paris Smaragdis. Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs. In *Proceedings of the International Conference on Independent Component Analysis and Blind Signal Separation ICA*, pages 494–499, Grenada, Spain, September 2004.

[186] Paris Smaragdis and Bhiksha Raj. Shift-invariant probabilistic latent component analysis. Technical Report TR2007-009, Mitsubishi Electric Research Laboratories, 2007.

[187] Paris Smaragdis, Bhiksha Raj, and Madhusudana Shashanka. Supervised and semi-supervised separation of sounds from single-channel mixtures. In *Proceedings of the International Conference on Independent Component Analysis and Signal Separation (ICA)*, pages 414–421, London, UK, September 2007.

[188] Paris Smaragdis, Bhiksha Raj, and Madhusudana Shashanka. Sparse and shift-invariant feature extraction from non-negative data. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 2069–2072, Las Vegas, Nevada, USA, March 2008.

[189] Carl Southall, Ryan Stables, and Jason Hockman. Automatic drum transcription using bi-directional recurrent neural networks. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 591–597, New York City, USA, August 2016.

[190] Carl Southall, Ryan Stables, and Jason Hockman. ADT with BLSTMP. 2018. submitted for publication.

[191] Vinícius M. A. Souza, Gustavo E. A. P. A. Batista, and Nilson E. Souza-Filho. Automatic classification of drum sounds with indefinite pitch. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, Killarney, Ireland, July 2015.

[192] Andrio Spich, Massimiliano Zanoni, Augusto Sarti, and Stefano Tubaro. Drum music transcription using prior subspace analysis and pattern recognition. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, Graz, Austria, September 2010.

[193] Dirk Van Steelant, Koen Tanghe, Sven Degroeve, Bernard De Baets, Marc Leman, and Jean-Pierre Martens. Classification of percussive sounds using support vector machines. In *Proceedings of the Annual Machine Learning Conference of Belgium and The Netherlands (BENELEARN)*, pages 146–152, Brussels, Belgium, January 2004.

[194] Dirk Van Steelant, Koen Tanghe, Sven Degroeve, Bernard De Baets, Marc Leman, and Jean-Pierre Martens. Support vector machines for bass and snare drum recognition. In *Classification – the Ubiquitous Challenge. Studies in Classification, Data Analysis, and Knowledge Organization*, pages 616–623. Springer, 2005.

[195] Bob L. Sturm. An analysis of the GTZAN music genre dataset. In *Proceedings of the International ACM Workshop on Music Information Retrieval with User-Centered and Multimodal Strategies MIRUM*, pages 7–12, Nara, Japan, October 2012.

[196] Nicolas Sturmel and Laurent Daudet. Signal reconstruction from STFT magnitude: A state of the art. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, pages 375–386, Paris, France, September 2011.

[197] Dennis L. Sun and Julius O. Smith III. Estimating a signal from a magnitude spectrogram via convex optimization. In *Proceedings of the Audio Engineering Society (AES) Convention*, San Francisco, USA, October 2012. Preprint 8785.

[198] Richard S. Sutton. Two problems with backpropagation and other steepest-descent learning procedures for networks. In *Proceedings of the Annual Conference of the Cognitive Science Society*, pages 823–831. Erlbaum, 1986.

[199] Hideyuki Tachibana, Nobutaka Ono, Hirokazu Kameoka, and Shigeki Sagayama. Harmonic/percussive sound separation based on anisotropic smoothness of spectrograms. *IEEE/ACM Transactions on Audio, Speech & Language Processing*, 22(12):2059–2073, 2014.

[200] Koen Tanghe, Sven Degroeve, and Bernard De Baets. An algorithm for detecting and labeling drum events in polyphonic music. In *Proceedings of the first MIREX*, pages 11–15, London, UK, 2005.

[201] Lucas Thompson, Simon Dixon, and Matthias Mauch. Drum transcription via classification of bar-level rhythmic patterns. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 187–192, Taipei, Taiwan, October 2014.

[202] Tijmen Tieleman and Geoffrey Hinton. RmsProp: Divide the gradient by a running average of its recent magnitude, October 2012.

[203] Adam Tindale, Ajay Kapur, George Tzanetakis, and Ichiro Fujinaga. Retrieval of percussion gestures using timbre classification techniques. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 541–545, Barcelona, Spain, October 2004.

[204] George Tzanetakis, Ajay Kapur, and Richard I. McWalter. Subband-based drum transcription for audio signals. In *Proceedings of the Workshop on Multimedia Signal Processing (MMSP)*, Shanghai, China, October 2005.

[205] Christian Uhle, Christian Dittmar, and Thomas Sporer. Extraction of drum tracks from polyphonic music using independent subspace analysis. In *Proceedings of the International Symposium on Independent Component Analysis and Blind Signal Separation (ICA)*, pages 843–847, Nara, Japan, April 2003.

[206] Marti Umbert, Jordi Bonada, Masataka Goto, Tomoyasu Nakano, and Johan Sundberg. Expression control in singing voice synthesis: Features, approaches, evaluation, and challenges. *IEEE Signal Processing Magazine*, 32(6):55–73, 2015.

[207] C. Gregor v. d. Boogaart and Rainer Lienhart. Note onset detection for the transcription of polyphonic piano music. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pages 446–449, New York City, USA, June 2009.

[208] Emmanuel Vincent. Improved perceptual metrics for the evaluation of audio source separation. In *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA)*, pages 430–437, Tel Aviv, Israel, March 2012.

[209] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte. Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1462–1469, 2006.

[210] Emmanuel Vincent, Nancy Bertin, Rémi Gribonval, and Frédéric Bimbot. From blind to guided audio source separation: How models and side information can improve the separation of sound. *IEEE Signal Processing Magazine*, 31(3):107–115, 2014.

[211] Tuomas Virtanen, Annamaria Mesaros, and Matti Ryynänen. Combining pitch-based inference and non-negative spectrogram factorization in separating vocals from polyphonic music. In *Proceedings of the ISCA Tutorial and Research Workshop on Statistical And Perceptual Audition (SAPA)*, pages 17–22, Brisbane, Australia, September 2008.

[212] Richard Vogl, Matthias Dorfer, and Peter Knees. Recurrent neural networks for drum transcription. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 730–736, New York City, USA, August 2016.

[213] Richard Vogl, Matthias Dorfer, and Peter Knees. Drum transcription from polyphonic music with recurrent neural networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 201–205, New Orleans, Louisiana, USA, March 2017.

[214] Thomas Völkel, Jakob Abeßer, Christian Dittmar, and Holger Großmann. Automatic genre classification on latin music using characteristic rhythmic patterns. In *Proceedings of the Audio Mostly : A Conference on Interaction with Sound*, Piteå, Sweden, September 2010.

[215] Stephen Voran. Exploration of the additivity approximation for spectral magnitudes. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, October 2015.

[216] DeLiang Wang. On ideal binary mask as the computational goal of auditory scene analysis. In P. Divenyi, editor, *Speech separation by humans and machines*, pages 181–197. Kluwer Academic, Norwell, Massachusetts, USA, 2005.

[217] Paul J. Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[218] Brian C. Wesolowski. *Testing a Model of Jazz Rhythm: Validating a Microstructural Swing Paradigm.* PhD thesis, University of Miami, Miami, Florida, USA, 2012.

[219] John Woodruff, Bryan Pardo, and Roger B. Dannenberg. Remixing stereo music with score-informed source separation. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 314–319, Victoria, Canada, October 2006.

[220] Chih-Wei Wu and Alexander Lerch. Drum transcription using partially fixed non-negative matrix factorization with template adaptation. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 257–263, Málaga, Spain, October 2015.

[221] Chih-Wei Wu and Alexander Lerch. Drum transcription using partially fixed non-negative matrix factorization. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, pages 1281–1285, Nice, France, September 2015.

[222] Chih-Wei Wu and Alexander Lerch. On drum playing technique detection in polyphonic mixtures. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 218–224, New York City, USA, August 2016.

[223] Chih-Wei Wu, Christian Dittmar, Carl Southall, Richard Vogl, Gerhard Widmer, Jason Hockman, Meinard Müller, and Alexander Lerch. A review of automatic drum transcription. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(9):1457–1483, 2018.

[224] Kazuyoshi Yoshii, Masataka Goto, and Hiroshi G. Okuno. Automatic drum sound description for real-world music using template adaptation and matching methods. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 184–191, Barcelona, Spain, October 2004.

[225] Kazuyoshi Yoshii, Masataka Goto, and Hiroshi G. Okuno. INTER:D: A drum sound equalizer for controlling volume and timbre of drums. In *Proceedings of the European Workshop on the Integration of Knowledge, Semantic and Digital Media Technologies (EWIMT)*, pages 205–212, London, UK, November 2005.

[226] Kazuyoshi Yoshii, Masataka Goto, and Hiroshi G. Okuno. Adamast: A drum sound recognizer based on adaptation and matching of spectrogram templates. *Annual Music Information Retrieval Evaluation eXchange (MIREX)*, 2005.

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

[228] Matthew D. Zeiler. ADADELTA: An adaptive learning rate method. *CoRR*, abs/1212.5701, 2012.

[229] Xinglei Zhu, Gerald T. Beauregard, and Lonce L. Wyse. Real-time signal estimation from modified short-time Fourier transform magnitude spectra. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(5):1645–1653, July 2007.

[230] Aymeric Zils, François Pachet, Olivier Delerue, and Fabien Gouyon. Automatic extraction of drum tracks from polyphonic music signals. In *Proceedings of the International Conference on Web delivering of Music (WEDELMUSIC)*, pages 179–183, Darmstadt, Germany, December 2002.