

COMPUTATIONAL METHODS
FOR TONALITY-BASED STYLE ANALYSIS
OF CLASSICAL MUSIC AUDIO RECORDINGS

Christof Weiß

geboren am 16.07.1986 in Regensburg

Dissertation

zur Erlangung des akademischen Grades
Doktoringenieur (Dr.-Ing.)

Angefertigt im:	Fachgebiet Elektronische Medientechnik Institut für Medientechnik Fakultät für Elektrotechnik und Informationstechnik
Gutachter:	Prof. Dr.-Ing. Dr. rer. nat. h. c. mult. Karlheinz Brandenburg Prof. Dr. rer. nat. Meinard Müller Prof. Dr. phil. Wolfgang Auhagen
Tag der Einreichung:	25.11.2016
Tag der wissenschaftlichen Aussprache:	03.04.2017

Acknowledgements

This thesis could not exist without the help of many people. I am very grateful to everybody who supported me during the work on my PhD. First of all, I want to thank Prof. Karlheinz Brandenburg for supervising my thesis but also, for the opportunity to work within a great team and a nice working environment at Fraunhofer IDMT in Ilmenau. I also want to mention my colleagues of the Metadata department for having such a friendly atmosphere including motivating scientific discussions, musical activity, and more. In particular, I want to thank all members of the Semantic Music Technologies group for the nice group climate and for helping with many things in research and beyond. Especially—thank you Alex, Ronny, Christian, Uwe, Estefanía, Patrick, Daniel, Ania, Christian, Anna, Sascha, and Jakob for not only having a prolific working time in Ilmenau but also making friends there.

Furthermore, I want to thank several students at TU Ilmenau who worked with me on my topic. Special thanks go to Prof. Meinard Müller for co-supervising my thesis, for a lot of scientific input, and for some very fruitful collaborations during my PhD work, but also for being always welcome in his great research group where I have now the honour of being part of. Thank you also Jonathan, Thomas, Stefan, Christian, Patricio, Frank, Julia, and Vlora for this pleasant working atmosphere and the good time in the past year.

I also received inspiration from other sides. In particular, I want to thank Simon Dixon, Matthias Mauch, and several PhD students from the Centre for Digital Music at Queen Mary University of London. Thank you for the chance to spend two extended research stays, which pushed me forward a lot. I am also grateful for all collaborations on the musicology side involving people in Würzburg, Saarbrücken, and others. In this context, I also want to thank Prof. Wolfgang Auhagen for co-supervising my dissertation. Furthermore, I am thankful to those who taught me to understand and love music, with a special mention of Karin Berndt-Vogel, Hermann Beyer, Prof. Zsolt Gárdonyi, Tobias Schneid, and Prof. Heinz Winbeck.

Fortunately, I had the opportunity to focus on my work and following my own ideas without being distracted with many other tasks. I am very grateful to the Foundation of German Business (Stiftung der Deutschen Wirtschaft), whose financial support during both my studies and my PhD time opened up many possibilities for me. Far more importantly, the unique spirit of this great community gave me a lot of inspiration. Thanks to everyone working in Berlin for creating this special atmosphere and thanks to all sdw friends in Würzburg, Ilmenau, Erfurt, and elsewhere for bringing this spirit to life.

Finally, I want to say a deep “thank you” to my parents Rita and Josef Weiß. Thank you for accompanying me all the way, for giving me the chance to do the things I want to do, and for laying the foundations to this. I also want to thank my brother Thomas and my extended family for all support and interest. And last, thank you, Ulli, for your love and support, and for being there in the good and in the harder times.

Abstract

With the tremendously growing impact of digital technology, the ways of accessing music crucially changed. Nowadays, streaming services, download platforms, and private archives provide a large amount of music recordings to listeners. As tools for organizing and browsing such collections, automatic methods have become important. In the area of *Music Information Retrieval*, researchers are developing algorithms for analyzing and comparing music data with respect to musical characteristics. One typical application scenario is the classification of music recordings according to categories such as musical *genres*.

In this thesis, we approach such classification problems with the goal of discriminating subgenres within *Western classical music*. In particular, we focus on typical categories such as historical periods or individual composers. From a musicological point of view, this classification problem relates to the question of *musical style*, which constitutes a rather ill-defined and abstract concept. Usually, musicologists analyze musical scores in a manual fashion in order to acquire knowledge about style and its determining factors. This thesis contributes with computational methods for realizing such analyses on comprehensive corpora of audio recordings. Though it is hard to extract explicit information such as note events from audio data, the computational analysis of audio recordings might bear great potential for musicological research. One reason for this is the limited availability of symbolic scores in high quality.

The style analysis experiments presented in this thesis focus on the fields of *harmony* and *tonality*. In the first step, we use signal processing techniques for computing *chroma representations* of the audio data. These semantic “mid-level” representations capture the pitch class content of an audio recording in a robust way and, thus, constitute a suitable starting point for subsequent processing steps. From such chroma representations, we derive measures for quantitatively describing stylistic properties of the music. Since chroma features suppress timbral characteristics to a certain extent, we hope to achieve invariance to timbre and instrumentation for our analysis methods.

Inspired by the characteristics of the chroma representations, we model in this thesis specific concepts from music theory and propose algorithms to measure the occurrence of certain tonal structures in audio recordings. One of the proposed methods aims at estimating the global key of a piece by considering the particular role of the final chord. Another contribution of this thesis is an automatic method to visualize modulations regarding diatonic scales as well as scale types over the course of a piece. Furthermore, we propose novel techniques for estimating the presence of specific interval and chord types and for measuring more abstract notions such as *tonal complexity*. In first experiments, we show the features’ behavior for individual pieces and discuss their musical meaning.

On the basis of these novel types of audio features, we perform comprehensive experiments for analyzing and classifying audio recordings regarding musical style. For this purpose, we apply methods from the field of machine learning. Using unsupervised clustering methods, we investigate the similarity of musical works across composers and composition years. Even though the underlying feature representations may be imprecise and error-prone in some cases, we can observe interesting tendencies that may exhibit some musical meaning when

analyzing large databases. For example, we observe an increase of tonal complexity during the 19th and 20th century on the basis of our features. As an essential contribution of this dissertation, we perform automatic classification experiments according to historical periods (“eras”) and composers. We compile two datasets, on which we test common classifiers using both our tonal features and standardized audio features. Despite the vagueness of the task and the complexity of the data, we obtain good results for the classification with respect to historical periods. This indicates that the tonal features proposed in this thesis seem to robustly capture some stylistic properties. In contrast, using standardized timbral features for classification often leads to overfitting to the training data resulting in worse performance. Comparing different types of tonal features revealed that features relating to interval types, tonal complexity, and chord progressions are useful for classifying audio recordings with respect to musical style. This seems to validate the hypothesis that tonal characteristics can be discriminative for style analysis and that we can measure such characteristics directly from audio recordings.

In summary, the interplay between musicology and audio signal processing can be very promising. When applied to a specific example, we have to be careful with the results of computational methods, which, of course, cannot compete with the experienced judgement of a musicologist. For analyzing comprehensive corpora, however, computer-assisted techniques provide interesting opportunities to recognize fundamental trends and to verify hypotheses.

Zusammenfassung

Im Zuge der fortschreitenden Digitalisierung vieler Lebensbereiche ist eine deutliche Veränderung des Musikangebots festzustellen. Streamingdienste, Downloadportale und auch private Archive stellen dem Hörer umfangreiche Kollektionen von Musikaufnahmen zur Verfügung. Bei der Strukturierung solcher Archive und der Suche nach Inhalten spielen automatische Methoden eine immer wichtigere Rolle. In diesem Kontext widmet sich der noch junge Forschungsbereich des *Music Information Retrieval* unter anderem der Entwicklung von Algorithmen und Werkzeugen zur inhaltsbasierten Suche, Navigation, Organisation und Analyse von Musikdatenbeständen. Eine typische Anwendung ist beispielsweise die Klassifizierung von Aufnahmen bezüglich bestimmter Kategorien wie beispielsweise musikalischer *Genres*.

Diese Arbeit befasst sich mit solchen Klassifikationsproblemen mit dem Ziel einer Differenzierung innerhalb der abendländischen Kunstmusik. Als typische Kategorien stehen dabei Epochen der Musikgeschichte oder einzelne Komponisten im Fokus. Aus musikwissenschaftlicher Sicht berührt diese Aufgabenstellung die Frage nach der *musikalischen Stilistik*, welche ein abstraktes und oft schwer definierbares Konzept darstellt. Bei der stilistischen Untersuchung führen Musikwissenschaftler typischerweise händische Partituranalysen durch, um Stilmerkmale in Musikstücken zu identifizieren. Ein wesentlicher Beitrag der vorliegenden Arbeit ist die Entwicklung computergestützter Methoden zur stilistischen Analyse umfangreicher Korpora von Audiodaten. Obwohl die Extraktion expliziter musikalischer Ereignisse wie Einzelnoten aus Audiodaten schwierig ist, kann die computergestützte Analyse von Audioaufnahmen eine Chance für die musikwissenschaftliche Forschung bieten, unter anderem weil qualitativ hochwertige Notentexte in symbolischer Kodierung oft nicht vorliegen.

Die stilistischen Untersuchungen in dieser Arbeit konzentrieren sich auf die Parameter *Harmonik* und *Tonalität*. Als erster Analyseschritt werden die Audiodaten mit Hilfe von Signalverarbeitungstechniken in *Chromadarstellungen* überführt. Diese semantischen “Mid-level”-Darstellungen spiegeln den harmonischen Gehalt der Musikaufnahmen im Bezug auf Tonhöhenklassen auf eine robuste Weise wider und stellen somit einen geeigneten Ausgangspunkt für weitere Verarbeitungsschritte dar. Aus diesen Chromadarstellungen werden dann unterschiedliche Merkmale zur quantitativen Beschreibung von Stilcharakteristika errechnet. Durch die Unterdrückung klangfarblicher Unterschiede in den Merkmalsdarstellungen wird eine Unabhängigkeit der Analysemethoden von der Klangfarbe und Instrumentation der Musik angestrebt.

Inspiziert von den Eigenschaften solcher Chromadarstellungen werden in dieser Arbeit musiktheoretische Konzepte aus den Bereichen Tonsatz beziehungsweise Harmonielehre modelliert und das Auftreten entsprechender tonaler Strukturen in den Audiodaten algorithmisch gemessen. Eine in dieser Arbeit eingeführte Technik dient der automatischen Analyse der Grundtonart eines Stückes unter Berücksichtigung der besonderen Rolle des Schlussakkords. Ein weiterer Beitrag ist eine automatische Methode zur Visualisierung von Modulationsstrukturen hinsichtlich diatonischer Skalen sowie von lokal vorherrschenden Skalentypen im Verlauf eines Stückes. Weiterhin führt diese Arbeit neue Algorithmen für die Messung von Intervall- und Akkordtypen sowie für die Quantifizierung abstrakter Konzepte wie der *tonalen*

Komplexität ein. Anhand einzelner Stücke werden zunächst die Eigenschaften der Merkmale aufgezeigt und ihre musikalische Bedeutung diskutiert.

Auf Grundlage dieser neu entwickelten Audiomerkmale werden umfangreiche Experimente zur Stilanalyse und Stilklassifizierung von Musikaufnahmen durchgeführt. Dabei kommen bekannte Algorithmen aus dem Bereich des maschinellen Lernens zum Einsatz. Mit Hilfe unüberwachter Lernmethoden (“unsupervised learning”) veranschaulicht diese Arbeit die stilistische Ähnlichkeit von Musikstücken im Bezug auf Komponisten und Kompositionsjahre. Obwohl die zugrunde liegenden Merkmalsdarstellungen im Einzelfall unpräzise und fehlerbehaftet sein können, lassen sich bei der Analyse größerer Datenmengen interessante Tendenzen beobachten, welche möglicherweise von musikgeschichtlicher Bedeutung sind. So lässt sich beispielsweise ein Anstieg der tonalen Komplexität im Verlauf des 19. und 20. Jahrhunderts auf Grundlage der vorgestellten Merkmale beobachten. Als wesentlicher Beitrag der Arbeit werden Experimente zur automatischen Klassifizierung von Musikdaten nach Epoche oder Komponist(in) durchgeführt. Auf zwei neu zusammengestellten Datensätzen werden bekannte Klassifikationsverfahren in Kombination sowohl mit tonalen Merkmalen als auch mit standardisierten Audiomerkmale getestet. Trotz der Vagheit der Aufgabenstellung und der Komplexität der Daten konnten gute Ergebnisse bei der Klassifikation nach Epochen erzielt werden. Die tonalen Merkmale scheinen dabei stilrelevante Eigenschaften auf eine stabile Art und Weise zu modellieren. Im Gegensatz dazu führt die Verwendung von Standardmerkmalen in Klassifikationsverfahren häufig zu einer Überanpassung der Modelle auf die Trainingsdaten, was sich negativ auf die Klassifikationsergebnisse auswirkt. Der Vergleich verschiedener tonaler Merkmale zeigt, dass Merkmale zur Beschreibung von Intervalltypen, tonaler Komplexität sowie von Akkordverbindungen geeignet für die Stilklassifizierung von Musikaufnahmen sind. Dadurch wird die Hypothese gestützt, dass sich tonale Eigenschaften in der Musik zur Stilunterscheidung heranziehen lassen und dass solche Eigenschaften direkt aus Audioaufnahmen gemessen werden können.

Zusammenfassend ist festzustellen, dass ein Wechselspiel zwischen den Disziplinen der Musikwissenschaft und der Audiosignalverarbeitung sehr vielversprechend sein kann. In der Anwendung auf Einzelfallbeispiele sind audiobasierte Analysemethoden kritisch zu hinterfragen und stehen sicherlich im Speziellen hinter der abwägenden Beurteilung durch einen Musikwissenschaftler zurück. Für den Vergleich von Musikstücken sowie die Betrachtung umfangreicher Korpora bieten die computergestützten Techniken jedoch interessante Möglichkeiten, um grundlegende Trends zu erkennen und Hypothesen zu verifizieren.

Table of Contents

Acknowledgements	iii
Abstract	v
Zusammenfassung	vii
Table of Contents	xi
1 Introduction	1
1.1 Contributions and Related Publications	4
1.2 Thesis Structure	7
2 Musicological Foundations	9
2.1 Tonality and Harmony	9
2.2 Tone, Pitch, and Pitch Class	11
2.3 Intervals	12
2.4 Tuning and Enharmonic Equivalence	14
2.5 Scales	17
2.6 Chords	21
2.6.1 Triads and Seventh Chords	21
2.6.2 Nonchord Tones	23
2.6.3 Functional Harmony and Chord Progressions	25
2.7 Key and Modulation	27
2.8 Models of Musical Pitch	30
2.8.1 Consonance and Dissonance	30
2.8.2 Geometric Pitch Models	31
2.9 Tonal Complexity	32
2.10 Tonality Aspects of Musical Style	32
3 Technical Foundations	35
3.1 Score Representations and Symbolic Data Types	35
3.2 Audio Representations	39
3.3 Spectrograms	41
3.4 Standardized Audio Features	44
3.5 Pitch-Based Features	48
3.5.1 Log-Frequency Spectrogram	48
3.5.2 Chroma Features	50
3.5.3 Timbre Invariance and Enhanced Chroma Features	53
3.5.4 Tuning Estimation	57
3.5.5 Temporal Resolution and Feature Smoothing	58
3.5.6 Properties of Chroma-Based Analysis	60

3.6	Machine Learning Methods	61
3.6.1	Experimental Design	61
3.6.2	Clustering	62
3.6.3	Classification	63
3.6.4	Dimensionality Reduction	65
4	State-of-the-Art	67
4.1	Overview	67
4.2	Global Key Detection	69
4.3	Local Key and Modulations	70
4.4	Recognition of Chords and Chord Progressions	71
4.5	Tonal Complexity	72
4.6	Classification and Clustering	72
4.6.1	Overview	72
4.6.2	Studies on Symbolic Data	73
4.6.3	Studies on Audio Data	75
5	Analysis Methods for Key and Scale Structures	77
5.1	Global Key Estimation Based on the Final Chord	77
5.1.1	Introduction	77
5.1.2	Proposed System	77
5.1.3	Evaluation	82
5.1.4	Conclusion	88
5.2	Local Estimation of Scales	89
5.2.1	Introduction	89
5.2.2	Musicological Foundations	90
5.2.3	Feature Extraction	91
5.2.4	Analysis of Modulations	91
5.2.5	Local Scale Type Estimation	96
5.2.6	Conclusion	100
6	Design of Tonal Features	103
6.1	Measuring Interval and Chord Categories	103
6.1.1	Introduction	103
6.1.2	Extraction of Chroma Features	103
6.1.3	Interval and Chord Features	105
6.1.4	Visualization Examples	107
6.1.5	Conclusion	108
6.2	Quantifying Tonal Complexity	109
6.2.1	Introduction	109
6.2.2	Musicological Implications	111
6.2.3	Proposed Method	112
6.2.4	Evaluation	116
6.2.5	Conclusion	120
7	Clustering and Analysis of Musical Styles	121
7.1	Dataset	121
7.2	Visualization of Audio Features through Music History	124

7.2.1	Data Mapping	124
7.2.2	Analysis of Chord Progressions	125
7.2.3	Analysis of Interval and Complexity Features	130
7.3	Style Analysis with Clustering Methods	132
7.3.1	Clustering Years	132
7.3.2	Clustering Individual Pieces	139
7.3.3	Clustering Composers	141
7.4	Conclusion	143
8	Subgenre Classification for Western Classical Music	147
8.1	Datasets	148
8.2	Dimensionality Reduction	150
8.3	Classification Experiments	155
8.3.1	Classification Procedure	155
8.3.2	Influence of the Classifiers	156
8.3.3	Influence of the Cross Validation Design	159
8.3.4	Influence of the Feature Types	161
8.3.5	Classification Results in Detail	166
8.4	Discussion	171
9	Conclusions	175
	Appendix	181
	Bibliography	185
	List of Figures	199
	List of Tables	201
	List of Abbreviations	203

1 Introduction

During the last decades, the ways of accessing and listening to music fundamentally changed. In the 1990s, the digital Compact Disc (CD) gained in popularity and gradually replaced prior analog media for storing music recordings. The invention of powerful audio compression technologies such as the MP3 format crucially influenced the distribution of digital recordings via the internet. With efficient storage technology, the enjoyment of music on portable digital devices (“MP3 players”) became popular. Recently, smartphones began to supersede such players more and more. Nowadays, music lovers often privately own large amounts of digital music recordings—up to several terabytes of data size. Public and commercial archives even surpass this size by several orders of magnitude. Beyond such locally stored recordings, online music streaming grew to a popular way of consuming music. Leading commercial suppliers provide several ten millions of songs to their customers.

With the growth of such archives, technologies for automatically searching, labeling, and organizing audio files have become important. Furthermore, automatic recommendation and selection of similar music plays a crucial role and led to business ideas such as “selling less of more” [7]. Often, the annotations and labels of the data are incomplete, inconsistent, or not useful for specific search criteria. Especially in private collections, we usually find many songs with purely technical labels such as “Track01.mp3.” Companies often make huge efforts to manually annotate and organize these files. In recent years, researchers proposed strategies towards an automatization of this annotation process by means of computer-based approaches. Starting from these contributions, the research area of Music Information Retrieval (MIR) evolved as a domain of growing importance. In particular, the International Society for Music Information Retrieval (ISMIR) emerged as an independent community. Contributions in this area are discussed—among others—at the annual ISMIR conference (since 2000).

Examples for typical MIR problems are the identification of recordings (Audio Fingerprinting) or artists. Other tasks are semantically more abstract such as browsing with musical queries (Query by Humming, Query by Example), or the search for cover songs and similar music. Furthermore, the automated extraction of musically relevant metadata such as the information on predominant instruments, tempo, location of beats and downbeats, musical key, chords, main melody, or the lyrics of a song play an important part. These tasks exhibit a high degree of interdependency since the extraction of meaningful metadata may again support the identification and search for similar music.

Beyond the identification of specific songs, automatic labeling of data with respect to more abstract categories may be useful. As an example, many researchers approached a problem known as **music genre classification** [46,237]. In such tasks, typical categories are so-called top-level *genres* such as Rock, Pop, Jazz, World music, or Classical. Since these terms are very vague and the genres often overlap with each other, genre classification constitutes a rather ill-defined problem. Beyond this, such categorization may be too superficial for specific purposes. Several publications approached a finer class resolution by considering subclasses of individual genres such as Rock [236], Electronic Dance Music [70], or Ballroom Dance Music [55]. Most of these methods mainly rely on timbral or rhythmic characteristics.

In this thesis, we focus on Western classical music. Thereby, our object of interest is the typical repertoire that dominates concert halls and classic radio programmes. When considering classical music as a “genre,” a subdivision becomes particularly important since this label usually comprises several centuries of music history, many different instrumentations, and various moods and purposes. There are only few methods addressing such subgenre classification for classical music. Apart from that, there are several ways to define subgenres. Some of the previous contributions used instrument categories as subclasses [225]. Such timbre-related subclasses are of importance since many listeners prefer music featuring certain instruments. For example, a listener may love piano music due to the sound of the piano but, at the same time, may dislike pieces featuring solo violin or opera arias by the same composer.

Nevertheless, a categorization of classical music into purely instrumental categories may not properly reflect the preferences of all listeners. Beyond the instrumentation, many classical music lovers generally prefer music by a certain composer—be it a piano sonata, a string quartet, or an opera. Furthermore, passionate listeners are often capable to identify the composer of a work after listening to only few measures—even if they cannot always explain the reasons for their decision. We conclude that there must be internal structures in the music that result in a composer-specific characteristic. Motivated by such observations, some researchers approached the identification of composers from audio data [98, 195]. Most of these previous studies mainly focused on a small number of composers since the task gets very complex for higher numbers and, moreover, some composers may be similar to each other with respect to musical style. Beyond this, considering individual composers may not be the only meaningful categorization. Rather, a listener may prefer music from a group of composers or a *historical period* in general. We may see this as a motivation to classify according to such periods (eras). A main contribution of this thesis is the development and evaluation of such subgenre classification systems for music recordings (Chapter 8). We want this classification to be invariant to timbre and instrumentation. For example, a Mozart piano sonata should obtain the same class label as a symphony or a string quartet since we assume some specific characteristics of Mozart’s pieces independently from the orchestration.

From the musicological point of view, the discussion of appropriate subgenres relates to the question of *musical style* and its definition. Even though musicologists have a good intuitive feeling of what style is, they argue about a clear definition of musical style and its determining factors. The notion of style is very ambiguous since it relates to secondary characteristics of music. Primarily, a composer usually aims at composing pieces each with an *individual* character—the *idea*—such as, for example, a new and catchy melody. In contrast, *style* rather relates to the way *how* a composer realizes this idea [19].

For analyzing composer styles, musicologists usually consider scores (sheet music). They manually identify structures such as specific chords or chord progressions that may be characteristic for the composer. Comparing the scores of various pieces by different composers, they obtain insights into the evolution and coherences of styles. Since this analysis by hand is cumbersome, musicologists often analyze a small number of representative piece and then generalize their findings to larger corpora. Here, computer-assisted methods may be helpful to support such claims with quantitative studies on a large amount of pieces. For approaching scores with computers, we need them to be explicitly encoded in **symbolic** formats. Concerning scores in graphical formats (images), we have to perform a conversion known as **Optical Music Recognition** (OMR). State-of-the-art OMR systems are still error-prone and require manual corrections.

Beyond musical scores, audio recordings of specific performances constitute another type of music representation. An audio recording captures the physical observation of such an interpretation (fluctuations of air pressure level) and, thus, represents the “sounding reality” of a musical piece in a specific performance. In this thesis, we address the analysis and classification of music on the basis of such audio recordings. This task is fundamentally different from score-based analysis. In the audio domain, we can only measure spectral energies over time and have no explicit encoding of note events. This makes the analysis of concrete musical structures a difficult task. For this reason, one might doubt whether audio recordings constitute a useful basis for analyzing musical styles.

There are some reasons why we think they may be helpful indeed. First, there is a practical argument. In many large music archives, pieces are only available in the form of audio recordings—even though there are some large score archives as well.¹ As we discussed in the beginning, audio is more relevant for many applications—such as browsing the archives of streaming services—since an audio recording itself constitutes the object of interest for a consumer. Second, scores may not capture all relevant properties of a musical piece. By itself, a score does not produce any sound. Interpreting that score adds many aspects that may be crucial for the music. Some scholars therefore proposed that “[...] we must identify every composition with its acoustical impression” [206]. Let us discuss this by considering an example. In an orchestral score, we may find a *forte* note for both flute and trumpet to be played at the same time. From the score, one would theoretically expect these notes to have equal loudness. However, in an acoustic realization, the (physically louder) trumpet tone may completely cover (mask) the flute tone, which may influence the perception of harmony, melody, or texture. A trained human—be it a musicologist or the composer—knows such effects when reading (or writing) the score of this piece. In contrast, computers do not. Generally, none of the representations of a piece—neither a score nor an audio recording of a specific performance—*is* that musical piece. Nevertheless, we assume that an audio recording may capture some important details of such a piece that we cannot easily find in a score.²

In principal, we could approach audio-based analysis by first detecting all note events and, thus, generating a score-like representation, which we could then analyze in the same way as score data. However, current state-of-the-art algorithms for this **automatic music transcription** task show poor performance compared to trained human experts. In particular, transcription systems are highly dependent on instrument characteristics. Because of that, we draw attention to more robust methods. For such purpose, *semantic mid-level representations* provide a good tradeoff between semantic meaning (“concreteness”) on the one hand and robustness to technical variations on the other hand. Regarding harmony and tonality—which we focus on in this dissertation—, **chroma representations** may fulfil these requirements. They only capture the pitch class information of the music over time while ignoring the musical octave of these pitches. Previous MIR research showed that chroma representations are able to capture tonal information in a way that is—to a certain extent—robust against timbral variation.

Ignoring the octave information crucially limits the possibilities of analyzing harmonic phenomena. Using chroma representations, we cannot discriminate an interval such as a perfect fifth from its complementary (a perfect fourth) since we lose this information on the

¹One example is the public International Music Score Library Project (<http://www.imslp.org>).

²For answering the questions what *is* music (or a musical piece), we would also have to consider the field of **music cognition**. From research in this area, we know that the *perception* of a performance fundamentally differs from the acoustic signal. **Perceptual audio coding** makes extensive use of such psychological phenomena for audio compression.

pitch class level. In Section 3.5.6, we discuss this in more detail. Because of these limitations, we focus on such musical concepts that refer to the pitch class level and, therefore, may be realized using chroma representations. For example, analyzing the use of a specific chord type such as the half-diminished seventh chord is, in general, possible with chroma features. In contrast, we cannot analyze its typical position (which chord note is the lowest). In Chapter 2, we provide an introduction of these music theory concepts and discuss their usability for chroma-based analysis.

In the subsequent chapters, we compare different types of chroma implementations with respect to timbre invariance and some kind of “musical meaning” (Section 3.5). We propose several algorithms to derive secondary features from chromagrams that may be useful for analyzing tonal and stylistic characteristic (Chapters 5 and 6). As an important aspect, these characteristics relate to various temporal scales of the music. One method serves to automatically detect the global key of a piece—generating a label such as “F♯ minor” (Section 5.1). Another algorithm aims at locally analyzing and visualizing the change of musical key throughout a piece (Section 5.2). Furthermore, we propose techniques for quantifying the use of certain interval and chord types or, more abstractly, the tonal complexity of the music on various time scales (Chapter 6). We discuss all of these methods by means of individual pieces and visually illustrate the features’ characteristics. Based on such automatically extracted descriptors, we perform several experiments for clustering and classifying music recordings with respect to stylistic properties. To identify meaningful style subgenres, we conduct some automatic clustering of pieces and composers and discuss the meaning of the results with respect to musical style (Chapter 7). For the classification according to historical periods and composers (Chapter 8), we compare our chroma-based system to a baseline method using standard spectrum-based features. We conduct several studies in order to evaluate the timbre invariance of the classification and to estimate the capability of our system to “learn” something that may be related to musical style.

1.1 Contributions and Related Publications

The majority of the results presented in this thesis were previously published [252, 254, 256–259]. In this section, we want to mention the main contributions of this dissertation and explain their relation to the corresponding publications. At the end of this section, we add a list of the relevant papers.

This thesis is an interdisciplinary work by touching the disciplines musicology, engineering, and informatics. Essentially, we approach questions from the field of musicology by using algorithmic methods. Our methods are inspired by and mainly relate to music theory concerning, in particular, theories on harmony and tonality. Since we deal with audio data, technologies from the signal processing domain play a decisive role. Thereby, one of our main contributions is the development of tonal audio features. In the final chapters, we apply techniques from the field of machine learning for clustering and classifying pieces on the basis of our features. From the results, we attempt to draw some conclusions on musical style.

Because of this interdisciplinary nature, we present both musicological and technical foundations as well as corresponding previous research including the following contributions:

- **An introduction to the musicological foundations of tonality.** In Chapter 2, we present and discuss the most important terms and concepts for tonal analysis. Most of the concepts originate from music theory. For several concepts, we introduce some

mathematical notation that we use in the subsequent chapters. This chapter is intended to serve as an introduction for researchers in the MIR field.

- **A compact overview of relevant techniques in audio and MIR research.** Chapter 3 provides a short summary of the audio processing basics and presents several standardized audio features. Furthermore, we give a more detailed overview of chroma extraction methods. This chapter also serves to fix some mathematical notation used in the subsequent chapters.
- **A literature review for related work in the MIR domain.** This state-of-the-art (Chapter 4) briefly summarizes the relevant work both for tonal analysis of audio recordings and for style classification of music data (symbolic and audio).

Concerning tonality analysis, this thesis contributes with several algorithms relating to different temporal resolutions and music theory concepts. This includes the following work:

- **A novel method for estimating the global key based on the final chord.** This key detection method is specifically suitable for classical music where a piece's final chord usually relates to the global key. In [252], we first proposed this algorithm together with an evaluation on three datasets. We re-compiled one of these datasets, which served as evaluation set in related work. For another public dataset (Saarland Music Data [169]), we created and published key annotations.³ In a Bachelor's thesis supervised by the author of this dissertation, Schaab [211] compared the performance of this method to state-of-the-art algorithms using an additional dataset. We further evaluated the impact of key detection performance for style classification with key-related features [211, 259]. We did not include these results in this thesis.
- **A novel method for analyzing local keys over the course of a piece.** This approach simplifies the key detection task to a 12-key problem by only considering diatonic scales. In [254], we showed that this can lead to robust and useful visualizations of the modulations in a piece. Furthermore, we extend the method for analyzing non-diatonic scale types. In a case study on H. Rott's first symphony, Habryka [83] discussed the benefits of such methods for musicological analysis. Beyond that, we tested the local key structures as basis for tonal segmentation of pop songs [253]. We do not consider these publications [83, 253] in this thesis. Furthermore, the Bachelor's thesis by Gräfe [80]—supervised by the author of this dissertation—presents an evaluation of classification experiments using local key properties for classifying music recordings with respect to era and composer categories. The results are not part of this dissertation.
- **A novel algorithm for deriving interval- and chord-related features from chromagrams.** We first published this idea in [256] where we tested the resulting features' efficiency for style classification. Beyond this application, this thesis provides a more profound discussion and visualization of the features.
- **A novel set of features relating to the tonal complexity of music on different time scales.** In [257], we made attempts towards defining notions of tonal complexity for our applications. Moreover, we presented realizations of such a quantification based on chroma vectors and visually analyzed these features' behavior for individual chords

³<http://www.mpi-inf.mpg.de/resources/SMD>

and whole movements. In [258], we added some more types of complexity features and tested their efficiency for classifying musical styles.

Beyond that, we performed several experiments to estimate these features' capability for capturing musical style characteristics. These experiments comprise the following contributions:

- **A novel dataset for analyzing and classifying styles in Western classical music.** The first dataset (*Cross-Era*) comprises each 400 pieces that are representative for the four historical periods Baroque, Classical, Romantic, and Modern (20th century). The pieces span a certain variety of composers and are balanced with respect to the instrumentation (200 pieces each for piano and orchestra). We provide comprehensive annotations as well as chroma-based features extracted from these audio files.⁴ Furthermore, we provide global key annotations for the 1200 pieces of the Baroque, Classical, and Romantic periods. Additionally, we compiled an add-on set (400 pieces), which comprises music from stylistically “transitional” composers. We used the *Cross-Era* dataset in several publications [256, 258, 259]. The full set *Cross-Era+Add-On* (2000 pieces) constitutes the basis for the clustering experiments presented in this thesis.
- **A novel dataset for evaluating composer identification tasks.** This *Cross-Composer* dataset contains each 100 pieces by the eleven composers J. S. Bach, L. van Beeethoven, J. Brahms, A. Dvořak, G. F. Handel, J. Haydn, F. Mendelssohn-Bartholdy, W. A. Mozart, J.-P. Rameau, F. Schubert, and D. Shostakovich. The pieces encompass a wide range of instrumentations and piece types. We published audio features and annotations for this dataset.⁵ The annotations include a detailed specification of the performing artists.
- **Visualizations and clustering results of the *Cross-Era+Add-On* dataset.** From this data, we extracted chord progressions with a publicly available algorithm. We proposed a method to illustrate audio features over the history based on the lifetime of the composers. For chord progression bigrams and tonal complexity features, we analyze the feature values regarding the historical time axis and discuss possible conclusions concerning the evolution of musical styles. Finally, we perform several clustering experiments on the basis of the mapped features (clustering years, pieces, and composers).
- **Classification experiments for style periods and individual composers.** Using the majority of features proposed in this work, we train and evaluate three machine learning classifiers for identifying the stylistic period (on *Cross-Era*) or the composer (on *Cross-Composer*) from audio recordings. We compare the performance against a baseline system relying on standard features. Furthermore, we investigate the robustness of classification results with respect to timbral variety and technical artifacts using a composer and an artist filter. We published similar experiments for *Cross-Era* in [256] (using interval and chord features) and in [258] (using tonal complexity features). In this thesis, we did not include the evaluation of key-related chroma histograms for classifying *Cross-Era* published in [259].

⁴<http://www.audiolabs-erlangen.de/resources/MIR/cross-era>

⁵<http://www.audiolabs-erlangen.de/resources/MIR/cross-comp>

In the following, we provide a chronological list of all publications that are relevant for this thesis:

- [252] Christof Weiß, “Global Key Extraction from Classical Music Audio Recordings Based on the Final Chord,” in *Proceedings of the 10th Sound and Music Computing Conference (SMC)*, 2013, pp. 742–747.
- [256] Christof Weiß, Matthias Mauch, and Simon Dixon, “Timbre-Invariant Audio Features for Style Analysis of Classical Music,” in *Proceedings of the Joint Conference 40th ICMC and 11th SMC*, 2014, pp. 1461–1468.
- [254] Christof Weiß and Julian Habryka, “Chroma-Based Scale Matching for Audio Tonality Analysis,” in *Proceedings of the 9th Conference on Interdisciplinary Musicology (CIM)*, 2014, pp. 168–173.
- [257] Christof Weiß and Meinard Müller, “Quantifying and Visualizing Tonal Complexity,” in *Proceedings of the 9th Conference on Interdisciplinary Musicology (CIM)*, 2014, pp. 184–187.
- [258] Christof Weiß and Meinard Müller, “Tonal Complexity Features for Style Classification of Classical Music,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015, pp. 688–692.
- [259] Christof Weiß and Maximilian Schaab, “On the Impact of Key Detection Performance for Identifying Classical Music Styles,” in *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, 2015, pp. 45–51.

1.2 Thesis Structure

This dissertation is structured as follows. The three chapters following this introduction provide foundations and previous research that are relevant for this thesis. Chapter 2 gives an overview of the relevant concepts in musicology and music theory regarding tonality analysis and its relation to musical style (Section 2.10). Moreover, we introduce mathematical notation to describe the relevant tonal structures. We particularly focus on concepts that refer to the pitch class level since we can realize them for audio data using chroma features. In Chapter 3, we discuss various types of music representations. For the audio domain, we outline the fundamental processing techniques such as the Short-Time Fourier Transform. In Section 3.4, we describe various types of standard spectrum-based audio features. Since chroma features play a decisive role in this thesis, we present more details on these features and discuss several chroma implementations and enhancement strategies (Section 3.5). Finally, we outline the main aspects of several machine learning methods used in the subsequent chapters (Section 3.6). Chapter 4 presents a literature review. We confine ourselves to mention the most important contributions for automatic tonality analysis of audio data as well as style classification studies for both symbolic and audio data.

Chapters 5–8 present the methods proposed in this thesis and their evaluation. In Chapter 5, we describe our novel method for global key detection relying on a piece’s final chord (Section 5.1). Furthermore, we propose a method for analyzing local keys and modulations based on diatonic scales as well as a more general analysis technique concerning scale types (Section 5.2). We visualize these results for a number of pieces throughout music history. Chapter 6 provides two novel strategies for deriving tonal features from chromagrams. The first method (Section 6.1) relates to the presence of interval and chord types. The second method (Section 6.2) serves to quantify tonal complexity on different temporal levels. We visualize the feature values for isolated chords and for the head movements of Beethoven’s sonatas. In Chapter 7, we introduce our style analysis dataset *Cross-Era+Add-On* and propose a method for mapping features onto a historical time axis. With this method, we

analyze chord progression bigrams (extracted with a public algorithm) and our complexity features over 300 years of music history (Section 7.2). We analyze both feature types using principal component analysis. On the basis of all features (chord bigrams and complexity), we perform unsupervised clustering experiments with respect to years, pieces, and composers (Section 7.3). Finally, Chapter 8 presents the results of our classification experiments. Beyond the *Cross-Era* dataset, we introduce in Section 8.1 a second dataset for composer identification (*Cross-Composer*). For chroma-based and standard features, we show visualizations of the feature space using Linear Discriminant Analysis (Section 8.2). We outline our classification procedure and discuss some details of cross validation (Section 8.3). The following sections show the results for different classifiers, cross validation settings, and feature constellations. We test the robustness of the classification systems to timbral variation and their capability for generalization to unseen data. For all classification experiments, we compare our chroma-based strategy to a standard spectrum-based system. Moreover, we look into the details of classification by investigating the types of errors (Section 8.3.5). Chapter 9 summarizes the results of this work and discusses the consequences of our findings. Furthermore, we give a perspective to future research directions.

2 Musicological Foundations

This chapter gives an overview of the fundamental terms and concepts for describing tonal phenomena in Western classical music. We expose these phenomena along with the most important ideas in music theory and the historical development of these ideas. For presentation, we display the concepts of tonality in common Western musical notation and assume the reader’s familiarity with the basic terms of music theory.¹ Furthermore, we introduce some mathematical modeling for the use in subsequent chapters.

State-of-the-art methods for computational audio analysis have shortcomings with respect to several qualities of tonality. For this reason, we put special emphasis on those concepts that one can adequately address on the basis of current signal processing techniques. Chapter 3 covers those limitations of current techniques that affect the description of tonal structures, along with the description of digital music representations.

For explanations of the basic musical terms, we follow the textbooks on harmony by Roig-Francolí [204], Kostka and Payne [122], and Laitz [127]. Several ideas link to Schönberg’s “Harmonielehre” [214] where the page numbers refer to the English translation by Carter [215]. Zsolt Gárdonyi’s and Hubert Nordhoff’s book [69]—only available in German—as well as Zsolt Gárdonyi’s lessons on music theory served as an inspiration to a number of concepts concerning the historical evolution of scales as well as the categorization of chord progressions. Some detailed information originates from Wikipedia articles.

2.1 Tonality and Harmony

There are a number of terms describing the organization of pitch as a musical dimension. Hereby, **tonality** is among the most prevalent ones but, at the same time, ambiguous and ill-defined. Although musicologists often ascribe this term to the french music theorist Fétis, his colleague Choron apparently used it first [226]. Among the numerous definitions existing in the literature, we choose a rather wide-ranging one: According to this concept, music is considered tonal when exhibiting a “systematic organization of pitch phenomena” [100]. This encompasses all music constructed of different pitches, including dodecaphonic and modal music.

Following a narrower but common definition, tonality denotes music’s property of featuring a referential pitch class or chord (“tonic”). Usually, the musical process resolves to that center at the end of a piece or section, thus generating a feeling of “arrival.” Schönberg emphasizes this formal aspect of tonality [215, p. 27]: “Tonality is a formal possibility [...], a possibility of attaining a certain completeness or closure.”² Examples for such kind of tonality are the major-minor tonality of the common practice period³, the modal systems of the prior Early music, or free modern systems that exhibit central tones that establish in a different way

¹See [122, 127, 204] for detailed explanations.

²At the same time, Schönberg does not consider the artistic use of this more specific tonality as an “eternal law.”

³In Western music history, the term “common-practice period” comprises the Baroque, Classical, and Romantic periods.

than in common-practice music.⁴ Terms such as “tonality” were proposed to describe this notion [199]. We refer to this as **referential tonality**, which serves as an umbrella term for tonal systems involving a reference pitch class.

One specific sample of such systems is the major-minor tonality of common-practice music—prevailing roughly from 1600 to 1910 while having a strong influence on the music beyond this period. In this tonal system, musical phenomena are organized around a referential tonic chord, which can be a major or a minor triad. The range of possible chords—assuming a twelve-tone temperament—led to the framework of 24 major and minor keys. Often, tonal music is considered as being restricted to this specific part of Western music. We stick to the general definition of tonality mentioned before and refer to the specific 24 key system as **major-minor tonality**. Within this system, the concept of a reference tonic chord entails “abstract relations that control melodic motion and harmonic succession over long expanses of musical time” and thus constitutes “the principal musical means with which to manage expectation and structure desire” [100]. Several theories cover the relation of pitches and chords towards the referential tonic chord [51, 200, 212, 249].

Out of this, one can see that tonality is a broader and more general concept than the less abstract terms harmony and melody. Hereby, **harmony** mainly relates to the “vertical” way of combining notes. When simultaneously sounding, groups of notes form some kind of entity in the listener’s mind—referred to as intervals (two notes) or chords (three or more notes). Furthermore, harmony comprises the succession of such musical constructs [45]. In contrast, **melody** covers the linear succession of notes in a monophonic consideration. Polyphonic textures—combinations of several monophonic lines—exhibit both harmonic and melodic aspects. Particular challenges arise when combining independent melodic lines. The field of **counterpoint** addresses these characteristics where voice leading rules play an important part.

Tonality is a hierarchical concept. On the one hand, it refers to different temporal scales—from the phrase level up to multi-movement works and work cycles. On the other hand, several concrete concepts describe tonal phenomena—pitch, pitch class, chord, scale, key, and more. They mutually interact in many ways. Over the history of music theory, scholars proposed several lines of argumentation to explain these terms and their interdependency. These theories either rest on acoustic properties of the tone [214], on the historical development of Western composition [69] or on theoretical and pedagogical reflections about chords [51, 52, 197, 200] or scales [212, 219, 249]. In the following sections, we introduce the fundamental terms. Starting with the characteristics of musical tones—overtones, pitch, and pitch class—(Section 2.2), we then introduce intervals (Section 2.3). We outline the problems of musical tuning and enharmonic equivalence (Section 2.4). Next, we describe musical scales (Section 2.5) before we present the concept of chords and functional harmony (Section 2.6). In Section 2.7, we cover the concept of key and modulations followed by the illustration of important pitch models (Section 2.8). Section 2.9 exposes some general thoughts on tonal complexity. In the final Section 2.10, we briefly discuss the impact of tonality for musical style analysis.

⁴As an example, we mention B. Bartok’s “Music for Strings, Percussion and Celesta”, which exhibits several of such central tones throughout each of the movements.



Figure 2.1. Harmonic series including the first 16 partials of C2. Using Western music notation, we can only approximate the exact pitches of the harmonics by rounding them to the equal-tempered scale (see Section 2.4). Please note the different indexing scheme when referring to “overtones” instead of “partials.”

2.2 Tone, Pitch, and Pitch Class

“The material of music is the tone: what it affects first, the ear” [215, p. 19]. Just as many music theorists, Schönberg considers the natural tone as the foundation of harmony. For representing such tones—produced by traditional pitched instruments or by the human voice—, we can use a series of sinusoids⁵ sounding simultaneously—the **partials**. As usual, we denote the lowest (first) partial of the tone as **fundamental**, the corresponding physical frequency as the **fundamental frequency** $f_0 \in \mathbb{R}^+$ given in Hertz (Hz). We refer to the higher partials as **overtones** with the first overtone corresponding to the second partial. For most musical instruments, the higher partials’ frequencies are close to integer multiples—the **harmonics** or **harmonic partials**—of the fundamental frequency.⁶ The frequency of the h -th harmonic partial $f_{\text{Part}}(h) \in \mathbb{R}^+$ is given as

$$f_{\text{Part}}(h) := h \cdot f_0 \quad (2.1)$$

for $h \in \mathbb{N}$. All partials together form the **harmonic series** of a musical tone. Figure 2.1 shows an approximate description of the harmonic series using Western music notation.⁷

For tones exhibiting partials that are harmonic to a certain extent, human listeners do not perceive these partials separately but as some kind of contribution to the tone. This psychoacoustic phenomenon leads to the perceptual concept of **pitch** that allows to order tones on a frequency-related logarithmic scale (“highness” of a tone [122]). The pitch information corresponds to the **perceived** fundamental frequency of a tone that may differ from the physical one because of inharmonicity effects. Moreover, the amount of oscillation energy in the fundamental may be considerably smaller than in (some of) the overtones without changing the pitch perception.

Due to the importance of overtones for pitch perception, humans rate tones as similar that share a high number of partials.⁸ Since we perceive pitch distances in a logarithmic sense, this effect is particularly prominent for pitches whose fundamental frequencies f_0^a and f_0^b

⁵To refer to a tone with a sinusoidal waveform, the term **pure tone** is common.

⁶Exceptions of this behavior occur for some pitched percussion instruments such as timpanies or tubular bells as well as for the low strings of the piano or the guitar. This phenomenon is called **inharmonicity**. In the following, we neglect such possible deviations of the partials from the harmonic frequencies and confine ourselves to only speak of *partials*.

⁷The exact frequencies of the harmonics differ from the ones indicated by the notation in Figure 2.1 depending on the tuning scheme assumed for notating the pitches. Section 2.4 outlines the detailed aspects of musical tuning.

⁸In particular, such ratings are made by listeners who are familiar with Western music. Researchers have shown that both for children [220] and listeners from non-Western cultures [114], in particular, the similarity of close pitches (on a logarithmic frequency scale) is of high importance, too.

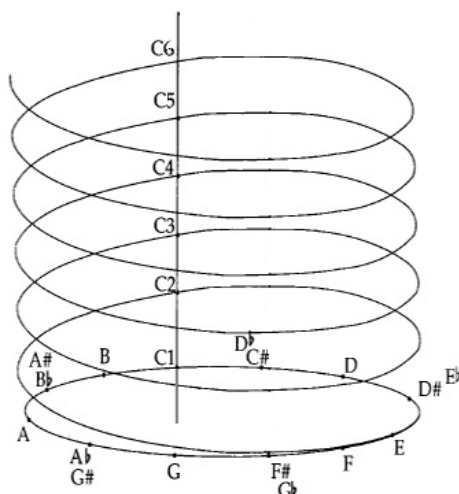


Figure 2.2. Shepard's helix of pitch perception. The height dimension illustrates the monotonically increasing tone height, the angular position refers to the circular notion of pitch class (image from [113]).

differ by powers of two:

$$f_0^b = 2^k \cdot f_0^a \Leftrightarrow \log_2 \left(\frac{f_0^b}{f_0^a} \right) = k \quad (2.2)$$

with $k \in \mathbb{Z}$. For $|k| = 1$, we call this an **octave** relation. Combining Equations (2.1) and (2.2), we obtain with $k = 1$

$$f_{\text{Part}}^b(h) = h \cdot f_0^b = 2h \cdot f_0^a = f_{\text{Part}}^a(2h). \quad (2.3)$$

Every second partial of the lower pitch (f_0^a) coincides with a partial of the higher pitch (f_0^b). To consider their similar quality, musicologists group pitches related by one or more octaves under the same **pitch class**. Roger Shepard's pitch helix (Figure 2.2) simultaneously illustrates the concepts of pitch class and pitch [223, 224]. Western music notation follows this principle when addressing pitches with a pitch class and an octave information. For instance, C4 denotes the pitch class C in the middle octave of the piano. With this octave labeling, we follow the international **scientific pitch notation**.

2.3 Intervals

Apart from the octave, the second most frequent pitch class in the harmonic series over f_0^a originates from the third partial. Similar to Equation (2.2), all pitches with a fundamental frequency f_0^b following the ratio

$$f_0^b = 3 \cdot 2^k \cdot f_0^a \quad (2.4)$$

with $k \in \mathbb{Z}$ belong to this pitch class. For a harmonic series over C, this is the pitch class G (see Figure 2.1). We call the distance between two pitches with a fundamental frequency relation of 3 : 2 a **fifth**.

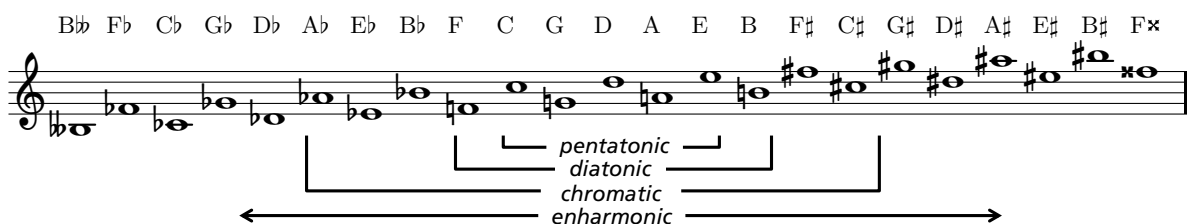


Figure 2.3. Pitch classes as a series of perfect fifths. We shifted the pitches to a suitable octave in order to ensure readability. The brackets indicate the pitch class content of four typical pitch class sets (scales).

<i>No. of Steps</i>	0	1	2	3	4	5	6	7
<i>Diatonic size</i>	1	2	3	4	5	6	7	8
<i>Generic name</i>	Unison	Second	Third	Fourth	Fifth	Sixth	Seventh	Octave

Figure 2.4. Generic intervals for the C major scale in relation to C4. The diatonic size specifies the distance in scale steps while counting equal pitches as 1. The interval names derive from the English or Latin words of the order number.

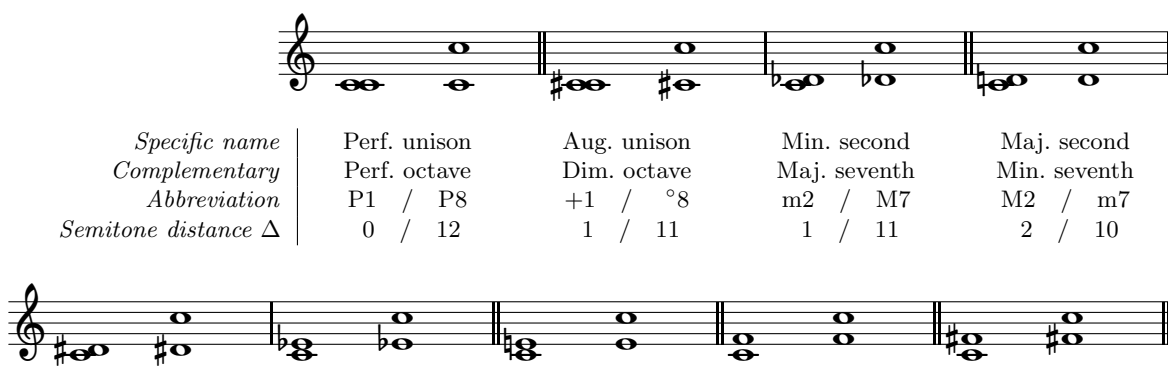
By iterating this relation, we can construct sets of pitch classes that play important roles in music history. A seven-part sub-sequence of this series of fifths forms the **diatonic scale**.⁹ Sub-sequences with different number of notes refer to other scales such as the **pentatonic scale** (five pitch classes) or the **chromatic scale** (twelve pitch classes) (see Figure 2.3). In Western music history, the seven-tone diatonic scale attained high importance since both the church modes of Early music and the (natural) minor and major scales share its structure—each with a different referential pitch class. Because of this scale’s predominance, the numbering of diatonic steps led to the traditional names of pitch distances—the **intervals**. Figure 2.4 illustrates these **generic intervals** [127]. Hereby, a **melodic interval** denotes the distance of successively played notes and can be ascending or descending while a **harmonic interval** refers to simultaneously sounding notes.

Intervals up to an octave are called **simple intervals**. Larger intervals sound similar to their simple counterparts, which we obtain by octave reduction. We therefore speak of **compound intervals** and refer to them as “octave + simple interval.”¹⁰ Some compound intervals have common names such as the ninth (octave + second) up to the thirteenth (octave + sixth). A similar concept—**inversion** of intervals—corresponds to an octave reduction of simple intervals (inverting the vertical pitch class order). We refer to the result as a **complementary interval**. A simple interval and its complementary sum up to an octave.

Western music notation evolved historically along with the pitch class content—up to reaching its current shape during the 17th century. This is why that system is particularly convenient for representing diatonic scales. We therefore obtain an interval’s generic name by counting the spaces and lines in the staff. Looking at Figure 2.3, we can extend the pitch class content to include more and different scales by using accidentals. These extended pitch class sets require a subtler discrimination of intervals. In the diatonic scale, a generic interval

⁹This observation is valid only when we map pitches onto the twelve-tone equal-tempered scale. In a detailed view, tuning aspects become important (see Section 2.4). For the historical construction of the diatonic scale, not only perfect fifths played a role but also the size of other intervals in the scale was optimized.

¹⁰Because of the strong similarity between compound intervals and their simple equivalent, we only explain further interval characteristics by means of simple intervals.



<i>Specific name</i>	Perf. unison	Aug. unison	Min. second	Maj. second
<i>Complementary</i>	Perf. octave	Dim. octave	Maj. seventh	Min. seventh
<i>Abbreviation</i>	P1 / P8	+1 / °8	m2 / M7	M2 / m7
<i>Semitone distance Δ</i>	0 / 12	1 / 11	1 / 11	2 / 10

Aug. second	Min. third	Maj. third	Perf. fourth	Aug. fourth
Dim. seventh	Maj. sixth	Min. sixth	Perf. fifth	Dim. fifth
+2 / °7	m3 / M6	M3 / m6	P4 / P5	+4 / °5
3 / 9	3 / 9	4 / 8	5 / 7	6 / 6

Figure 2.5. Specific names of intervals and their complementaries. The modifiers specifying the exact size are “perfect” (perf.), “major” (maj.), “minor” (min.), “diminished” (dim.), and “augmented” (aug.). The table’s third row shows a common abbreviation as specified in [204]. The last row gives the distance in semitones Δ referring to the equal-tempered scale.

may refer to multiple frequency relations as soon as we consider all scale notes as a possible reference pitch. We define an interval’s exact size with an additional **modifier** obtaining the **specific** interval name. Traditionally, we characterize octave and fifth as well as their complementary intervals as **perfect**, the other intervals as **major** or **minor**. Furthermore, all generic interval types can appear in augmented and diminished versions. Figure 2.5 illustrates the specific names of the intervals along with their complementary equivalents. Different versions of a generic interval share the diatonic number but not the frequency relation. This is why the diatonic scale does not constitute an equally spaced division of the octave but contains both whole steps (major seconds) and half steps or semitones (minor seconds).

We derived the intervals from the diatonic scale in order to understand the Western naming convention. Just as we explained the fifth, we also can deduce other intervals from the harmonic series (Figure 2.1). This leads, for example, to a major third with a frequency relation of 5 : 4 or to a minor third of 6 : 5. For these **pure intervals**, several harmonics of the two pitches coincide.

2.4 Tuning and Enharmonic Equivalence

During the Early music periods, the pitch content in use evolved from one diatonic scale towards including further scales that relate by a horizontal shift in Figure 2.3. With the increasing use of keyboard instruments during the 17th century, this led to a central problem in Western harmony—the conflict between the frequencies of the natural overtones and an equal division of the octave for obtaining similar steps between scale degrees. Around the 17th century, several theorists proposed tuning systems for keyboard instruments to approach this problem—such as the meantone temperament by Gioseffo Zarlino based on pure major thirds with a frequency ratio of 5 : 4. Another example is the Pythagorean tuning based on pure perfect fifths (3 : 2). We refer to these tuning systems based on pure intervals as **just intonation**. In such systems, some intervals have nice frequency ratios. On the downside, some other intervals appear to be seriously detuned leading to unusable scales and intervals

on the keyboard. For this reason, Andreas Werckmeister, Johann Kirnberger, and others proposed so-called **well-tempered** tuning systems, which allow to play scales based on all twelve chromatic pitches without considerably mistuned intervals. The strict realization of this idea leads to the **twelve-tone equal temperament** of today’s keyboard instruments¹¹ where the octave (2 : 1) is divided into twelve semitones with an equal step size of

$$f_0^b = \sqrt[12]{2} \cdot f_0^a \quad \Leftrightarrow \quad \log_2 \left(\frac{f_0^b}{f_0^a} \right) = \frac{1}{12}. \quad (2.5)$$

Using this scale, a pitch class is considered coincident with its enharmonic counterpart shifted by twelve fifth intervals (Figure 2.3). Hence, we have to reduce these fifth intervals by 1/12 of the **Pythagorean comma**

$$\frac{(3/2)^{12}}{2^7} \approx 1.0136. \quad (2.6)$$

That is, G \sharp in Figure 2.3 is about $\log_2(1.0136) \cdot 1200 \approx 23.5$ **Cent** (percent of an equal-tempered semitone) higher than the corresponding A \flat when tuned according to a series of perfect fifth intervals (Pythagorean tuning). In equal temperament, the Pythagorean comma splits up equally over the twelve fifths. Therefore, the equal-tempered version of the perfect fifth is by approximately two Cent lower than the pure version. Similarly, there is a difference of about 21.5 Cent between a pure major third and four concatenated perfect fifths—the **syntonic comma**. Because of such differences, the harmonic partials of a note do not perfectly match other notes within an equal-tempered scale—in contrast to the notation in Figure 2.1.

In the twelve-tone equal temperament, the chromatic scale in Figure 2.3 closes to a circle so that the altered pitch classes coincide:

$$G\sharp \hat{=} A\flat, \quad D\sharp \hat{=} E\flat, \quad F\sharp \hat{=} G\flat, \quad B\sharp \hat{=} C, \quad F\flat \hat{=} E, \quad \dots \quad (2.7)$$

We refer to this observation as **enharmonic equivalence**. This corresponds to the piano’s key arrangement with twelve keys per octave. Since numerous harmonic phenomena derive from diatonic scales—as well as the Western notation system—, enharmonic spelling of pitches in scores constitutes an important issue in order to ensure readability. Especially for Early music and Baroque music, musicians usually consider pitch spelling for intonation—such as players of wind or string instruments, or singers. As we outline in Section 3.5.6, we do not resolve these subtle pitch differences with our analysis method. Instead, we always assume the pitch class content of the twelve-tone equal-tempered scale.

Apart from such local **microtuning** aspects, we need to consider a **global tuning**. By tradition, musicians use the middle A4 as reference pitch (**concert pitch**). The frequency assigned to the concert pitch increased over the eras with today’s standard value of

$$f_{\text{concert}} := 440 \text{ Hz}. \quad (2.8)$$

Nowadays, interpreters sometimes adjust the concert pitch to lower values following the results of historical research. A common value for historical performance practice is

$$f_{\text{concert}}^{\text{hist}} := 415 \text{ Hz}, \quad (2.9)$$

¹¹On the piano, this is not exactly true since the inharmonicity of the low strings requires a pitch correction. On the organ, historical tunings are still in use to enable historically faithful interpretations.

which is close to the pitch $A\flat$ in 440 Hz tuning.

According to the observations presented in this section, we formalize pitch as a simple numbering of the equal-tempered scale:

$$p \in [0 : 127] := \{0, 1, \dots, 127\} \subset \mathbb{N}_0 \quad (2.10)$$

with $p = 60$ corresponding to $C4$. We obtain the following relation between pitch and fundamental frequency:

$$f_0(p) = 2^{(p-69)/12} \cdot f_{\text{concert}}. \quad (2.11)$$

Similarly, we refer to the pitch class of a note as a number

$$q \in [0 : 11]. \quad (2.12)$$

In our notation, $q = 0$ denotes the pitch class C leading to the correspondence

$$(0, 1, \dots, 11) \cong (C, C\sharp, \dots, B). \quad (2.13)$$

Since $p = 0$ refers to a tone with pitch class C , we obtain the following relation:

$$q(p) = p \bmod 12 \quad (2.14)$$

For the octave number in scientific pitch notation, we obtain

$$u(p) = \lfloor p/12 \rfloor - 1. \quad (2.15)$$

Thus, the pitch derives from pitch class and octave number as

$$p(q, u) = q + 12 \cdot (u + 1). \quad (2.16)$$

As for the pitches, enharmonic equivalence affects intervals as well. Two intervals are enharmonically equivalent when they have the same semitone distance Δ in the equal-tempered scale:

$$+1 \cong m2, \quad M2 \cong \circ 3, \quad +2 \cong m3, \quad +3 \cong P4, \quad \dots \quad (2.17)$$

We define a melodic interval between two pitches p^a and p^b as the distance

$$\Delta(p^a, p^b) = p^b - p^a \quad (2.18)$$

whereas, for harmonic intervals, only $|\Delta(p^a, p^b)|$ is relevant. That way, we can avoid all diminished or augmented intervals in Table 2.5 except for the augmented fourth (also referred to as “tritone”). For a compound interval, we obtain the corresponding simple interval by

$$\Delta_{\text{simple}} = \Delta_{\text{compound}} \bmod 12. \quad (2.19)$$

The complementary interval relates to its original counterpart via

$$\Delta_{\text{complementary}} = 12 - \Delta_{\text{original}}. \quad (2.20)$$

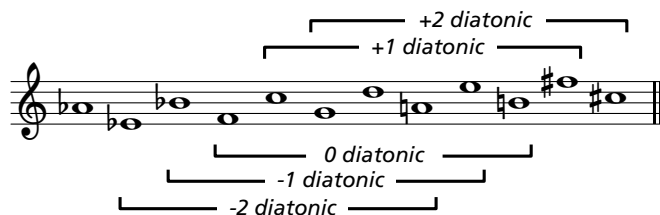


Figure 2.6. Chromatic scale in a perfect fifth ordering. The brackets are marking several diatonic sub-sequences. We name the scales according to the number and type of accidentals necessary in Western music notation (absolute fifth measurement). For example, the +1 diatonic scale requires one sharp ($F\sharp$), the +2 diatonic scale requires two flats ($B\flat$, $E\flat$). The 0 diatonic scale encompasses the white keys on a piano. Diatonic scales with a close relation share a high number of pitch classes.

2.5 Scales

We already introduced the diatonic and chromatic scales. In general, we define a **scale** as a set of pitch classes with a referential pitch class that exhibits a particularly emphasis and “stability” compared to the other pitches in the scale. In that understanding, both diatonic scale and chromatic scale are rather *scale families* than individual scales since they do not exhibit a reference pitch class. These scale families can provide the pitch class material for a certain section of music.

Regarding diatonic scales, several transpositions are possible. For the relation of these transpositions, simple ratios of fundamental frequencies play an important role—corresponding to lower partials in the harmonic series. As a consequence, fifth-related diatonic scales seem to be more harmonically similar than scales shifted by a small interval. Because of the perfect fifth structure of the diatonic scale, those fifth-related scales share a high number of common pitch classes (six out of seven). Following [69], we refer to a diatonic pitch class set by specifying the number $d \in \mathbb{Z}$ of sharp (“+”) or flat (“−”) accidentals required for notation (**absolute fifth measurement**). From this, we can compute a distance $\mathcal{D} \in \mathbb{Z}$ between diatonic scales (**relative fifth measurement**). For instance, the distance between two scales with $1\sharp$ ($d = 1$) and $3\flat$ ($d = -3$), respectively, is:

$$\mathcal{D}(+1, -3) = (-3) - (+1) = -4. \quad (2.21)$$

For details, see Figure 2.6.

In Gregorian chant and Renaissance vocal polyphony, all notes of the diatonic pitch class set served as reference note (**finalis**)—with one exception (the Locrian scale). The most common scales—known as **church modes**—are the upper four in Figure 2.7 (a–d). Named after Greek and Asian regions, Dorian (Mode I), Phrygian (Mode III), Lydian (Mode V), and Mixolydian (Mode VII) form the basis of the ancient modal system. Furthermore, there are derived versions differing only in the typical melodic structure (Modes II, IV, VI, VIII). In his “Dodecachordon” (1547), Glarean introduced the additional modes Aeolian and Ionian with their derivatives (modes IX–XII). They constitute the basis for the major-minor tonality of the common-practice period. For later music, the modes gained in importance again—particularly for late Romantic and impressionist music as well as for jazz improvisation.

For major-minor tonality, the most important scale is the **major scale** equaling the Ionian mode. We illustrate its detailed properties in Figure 2.8. Hereby, the caret numbers $\hat{1}$, $\hat{2}$ etc. denote the scale degrees in relation to the reference pitch. In contrast, the pedagogical

a) Dorian

b) Phrygian

c) Lydian

d) Mixolydian

e) Aeolian ($\hat{=}$ natural minor scale)

f) Locrian

g) Ionian ($\hat{=}$ major scale)

Figure 2.7. Diatonic modes. On the left hand side, we display the modes as diatonic shifts of the 0 diatonic scale without accidentals. On the right hand side, we show the same scale type over C as reference pitch class.

concept of **relative solmisation**¹² assigns constant **solfège syllables** to each note of a diatonic scale set. In the 0 diatonic scale, for example, the pitch class C always obtains the syllable *do* (see Table 2.1) independently from the chosen reference note. As we mentioned before, the major scale is not equally spaced in pitch with respect to the twelve-tone equal-tempered scale. Between the scale degrees $\hat{3}$ – $\hat{4}$ as well as $\hat{7}$ – $\hat{8}$ (corresponding to $\hat{7}$ – $\hat{1}$), a half step (H) or semitone occurs. The remaining steps are whole steps (W). The positions of the half steps circularly shift for the other diatonic modes. Therefore, *mi–fa* always forms a half step whereas the size of $\hat{3}$ – $\hat{4}$ depends on the specific scale. Additionally, there are common functional names for the scale degrees such as “tonic,” “mediant,” or “leading tone” (Figure 2.8). They behave in the same way as the scale degree numbers introduced previously. Here, it is important to avoid confusions between the scale degrees as pitch classes and other harmonic structures—such as chords or other scales—built upon these pitch classes. Therefore, we use a more specific reference such as “tonic note.”

For the **minor scale**, we find a different situation (Figure 2.9). In the aeolian scale—also called **natural minor**—, $\hat{7}$ – $\hat{8}$ results in a whole step. To preserve the harmonic quality of the raised leading tone as in the major scale, we alter this tone to $\sharp\hat{7}$ obtaining the **harmonic**

¹²This concept (also called “movable do solfège”) is not to confuse with the absolute pitch spelling (“fixed do solfège”) used in Romance languages. Besides the diatonic notes, there are also syllables for alterations such as *fi* for the raised *fa* degree or *ti* for the flatted *ti*.

Degree	$\hat{1}$	$\hat{2}$	$\hat{3}$	$\hat{4}$	$\hat{5}$	$\hat{6}$	$\hat{7}$	$\hat{8} = \hat{1}$
Name	Tonic	Supertonic	Mediant	Subdominant	Dominant	Submediant	Leading Tone	Tonic
Solfège	do	re	mi	fa	sol	la	ti	do

Figure 2.8. C major scale with scale degree numbers. Between the degrees $\hat{3}$ – $\hat{4}$ and $\hat{7}$ – $\hat{8}$ (equals $\hat{7}$ – $\hat{1}$ when referring to pitch classes), a half step occurs (m2). All other steps are whole steps (M2). The second column in the table lists the functional names of the scale degrees. In the last row, we show a common version of the solfège syllables used for relative solmization of scale degrees.

Table 2.1. Solfège syllables for the scale degrees of the diatonic modes.

Scale Degree	$\hat{1}$	$\hat{2}$	$\hat{3}$	$\hat{4}$	$\hat{5}$	$\hat{6}$	$\hat{7}$
Ionian	do	re	mi	fa	sol	la	ti
Dorian	re	mi	fa	sol	la	ti	do
Phrygian	mi	fa	sol	la	ti	do	re
Lydian	fa	sol	la	ti	do	re	mi
Mixolydian	sol	la	ti	do	re	mi	fa
Aeolian	la	ti	do	re	mi	fa	sol
Locrian	ti	do	re	mi	fa	sol	la

minor scale.¹³ This leads to the unusual interval of an augmented second (+2) between $\hat{6}$ – $\hat{7}$. To solve this melodic problem, we alter the submediant as well ($\sharp\hat{6}$) to obtain a smoother melodic interval. This generates the upward version of the **melodic minor** scale. For downward melodic movement, both alterations ($\sharp\hat{6}$ and $\sharp\hat{7}$) are not common. This leads to a larger set of nine pitch classes and, thus, a more complicated situation for minor scales.

Besides the diatonic scale types, there are other scales based on fifth relations. The pentatonic scale is a five-part sub-sequence of the series of fifth (see Figure 2.3) and plays an important role in impressionist music. Other scales do not form a consecutive excerpt of the fifth series. One example is the acoustic scale, which is relevant for a number of 20th century compositions. We can derive this scale from the harmonic series by selecting the first seven pitch classes. It is similar to the major scale but contains $\sharp\hat{4}$ and $\flat\hat{7}$ as alterations. There are also scales constructed from a symmetrical division of the octave assuming the equal-tempered scale as basis. One example is the six-note whole tone scale. We find other symmetrical divisions for the octatonic scale (half and whole steps alternating)—also called diminished scale—and the hexatonic scale (half steps and minor thirds alternating). Figure 2.10 illustrates examples for such scales.

For the non-diatonic scales, the common notation system does not provide an ideal representation. When considering scales as pitch class sets

$$S \subset [0 : 11] \quad (2.22)$$

we better see the symmetry of, e.g., the whole tone scale:

$$S_{\text{Wholetone}} = \{0, 2, 4, 6, 8, 10\}. \quad (2.23)$$

¹³We discuss the reasons for this alteration in Section 2.6.

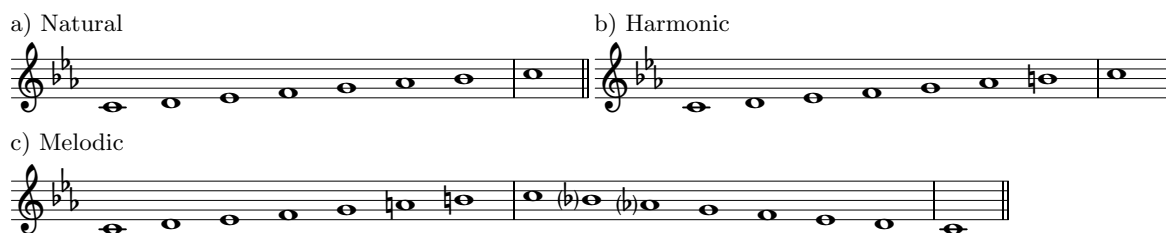


Figure 2.9. Different versions of the C minor scale. We notate the scales using the key signature of C minor. For indicating the alterations, we place accidentals next to the notes.

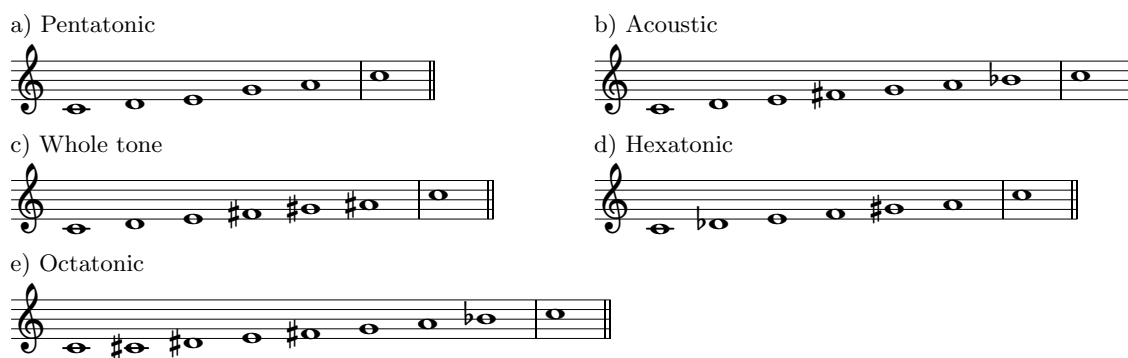


Figure 2.10. Several non-diatonic scales based on C. For the symmetrical scales (c–e), the traditional notation system is not convenient. For example, it does not reflect the equidistant spacing of the whole tone scale.

Alternatively, we can model a pitch class set as an “activation vector” or “energy distribution” $\mathbf{T} \in \mathbb{R}^{12}$ for the twelve chromatic pitch classes. Then, a specific pitch class q can be part of the scale ($T_q = 1$) or not ($T_q = 0$). For the whole tone scale, we obtain

$$\mathbf{T}^{\text{Wholetone}} = (1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0)^{\text{T}}. \quad (2.24)$$

This better shows the symmetry of such scales. The other scales introduced in this chapter correspond to the following pitch class vectors:

$$\begin{aligned} \mathbf{T}^{\text{Chromatic}} &= (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)^{\text{T}} \\ \mathbf{T}^{\text{Diatonic}} &= (1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1)^{\text{T}} \\ \mathbf{T}^{\text{NaturalMinor}} &= (1, 0, 1, 1, 0, 1, 0, 1, 1, 0, 1, 0)^{\text{T}} \\ \mathbf{T}^{\text{HarmonicMinor}} &= (1, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1)^{\text{T}} \\ \mathbf{T}^{\text{MelodicMinor}} &= (1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1)^{\text{T}} \\ \mathbf{T}^{\text{Pentatonic}} &= (1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0)^{\text{T}} \\ \mathbf{T}^{\text{Acoustic}} &= (1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 0)^{\text{T}} \\ \mathbf{T}^{\text{Hexatonic}} &= (1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0)^{\text{T}} \\ \mathbf{T}^{\text{Octatonic}} &= (1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0)^{\text{T}} \end{aligned} \quad (2.25)$$

This representation also helps to recognize half steps and whole steps. We will present further ideas relating to pitch class sets in Section 2.8.2. In Equation (2.25), the pitch class vectors

<i>Thirds</i>	(M3, m3)	(m3, M3)	(m3, m3)	(M3, M3)
<i>Frame interval</i>	P5	P5	$^{\circ}5$	+5
<i>Triad type name</i>	Major	Minor	Diminished	Augmented
<i>Abbreviation</i>	M	m	$^{\circ}$	+

Figure 2.11. Basic triad types above C4. Here, we show the triads in in root position (tertian structure).

refer to scales on the pitch class C. To obtain a transposed¹⁴ version $\tilde{\mathbf{T}}$ based on the pitch class $q^{\text{ref}} \in [0 : 11]$, we have to perform a circular shift of the vector entries:

$$\tilde{T}_q = T_{(q+q^{\text{ref}}) \bmod 12} \quad (2.26)$$

with $q \in [0 : 11]$.

2.6 Chords

2.6.1 Triads and Seventh Chords

Just as we consider scales as the “imitation of the tone on the horizontal plane” (“analysis of the tone”), chords constitute the analogue “on the vertical” plane (“synthesis” of the tone) [215, p. 26]. Western music grounds on monophonic chant. Later, composers combined more and more horizontal lines simultaneously (vocal polyphony). Thereby, chords occurred as events of coinciding notes while strictly following rules of harmony and counterpoint. With the beginning 17th century, these chords assumed a separate existence due to the arising monody and the basso continuo. From this era on, the “vertical” understanding of note groups particularly influenced composition and harmony analysis. According to this “chordal” perception of music, chords comprising three or more notes constitute the basic harmonic unit; harmonic intervals are components of chords rather than their origin [137].

Similar to the tone—as a compound of partials—, humans perceive chords as an entity rather than as individual notes. The most frequent chords are **triads**. The **major triad** (M) consists of three pitches, e.g., C4, E4, and G4. The major triad’s pitch classes correspond to the first three pitch classes that contribute to the harmonic series. Because of that, humans perceive this chord as a stable sound. In terms of intervals, the major triad constitutes a tuple of two thirds (M3, m3) where the outer notes form a P5. Because of the high stability of the perfect fifth interval, the **minor triad** (m3, M3) behaves stable as well—though the pitch class of the m3 above the root note is none of the lower partials. Concatenating twice the same third interval, we obtain the **diminished** and the **augmented** triad, named after their frame interval’s quality ($^{\circ}5$ or +5). Figure 2.11 shows these basic triads.

In the **tertian** structure—built out of concatenated thirds—, we refer to the triad’s constituent notes as **root**, **third**, and **fifth**. A triad is in **root position** when the root note is lowest (the bass note). For the **inversions** of triads, either the third note (first inversion or 6 **chord**) or the fifth note are lowest (second inversion or 4 **chord**).¹⁵ Due to the structure

¹⁴Here, we refer to the “musical” transposition, which corresponds to a shift in pitch by a constant interval.

¹⁵This **figured bass** notation practice stems from the Baroque period. Together with a notated bass line, additional numbers indicate the chord notes as intervals above the bass note (not the root!). Accidentals next to the numbers denote alterations of the chord notes. The numbers 3 and 5 may be absent.

<i>Inversion</i>	Root pos.	1st inv.	2nd inv.	1st inv. (open pos.)
<i>Figured bass notation</i>	($\overset{5}{3}$)	6	$\overset{6}{4}$	6
<i>Bass note</i>	Root	Third	Fifth	Third

Figure 2.12. Triad inversions shown for the CM triad. The last chord is in open position. Such detailed aspects of pitch arrangement (voicing) do not affect other harmonic properties of a chord such as chord type or inversion.

of the harmonic series, major and minor triads in root position are more stable than their inversions. This leads to a different harmonic usage of inverted triads. When the size of all intervals between chord notes is minimal¹⁶, we speak of **close position**. All other arrangements of notes (**voicings**) are in **open position**. These voicing aspects only slightly affect the harmonic quality of sounds as long as the bass note belongs to the same pitch class.

When referring to chords as a more abstract notion, we can think of them as sets of pitch classes sounding simultaneously. For the major triad based on C, we obtain the three-part set

$$S_{\text{CM}} = \{0, 4, 7\} \quad (2.27)$$

and the activation vector

$$\mathbf{T}^{\text{CM}} = (1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0)^{\text{T}}. \quad (2.28)$$

Note that these representations are invariant under triad inversions and octave shifts of any chord note but not under transposition. Equation (2.28) also describes the CM⁶ and CM⁶₄ chords. However, to specify a DbM chord, we need to perform a circular shift.

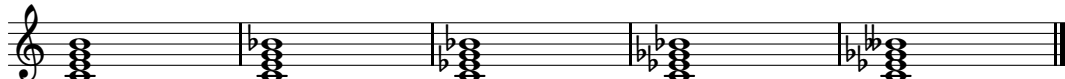
The other chord types correspond to the following pitch class vectors (based on C):

$$\begin{aligned} \mathbf{T}^{\text{Cm}} &= (1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0)^{\text{T}} \\ \mathbf{T}^{\text{C}^\circ} &= (1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0)^{\text{T}} \\ \mathbf{T}^{\text{C}^+} &= (1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0)^{\text{T}} \end{aligned} \quad (2.29)$$

Extending the tertian structure to four-note structures, we obtain **seventh chords**—since three concatenated thirds result in a seventh interval. We can define seventh chords as triples of thirds such as (M3, m3, m3)—or pairs of a triad and a specific seventh interval above the root note (M, m7). Over the course of the 17th century, seventh chords obtained an independent role. For instance, J. S. Bach made considerable use of the diminished seventh chord. In classical harmonic, the dominant seventh chords is of major importance. Romantic harmony extensively features seventh chords such as the half-diminished one (R. Wagner and others). Figure 2.13 displays some commonly used seventh chord types. As for the triads, seventh chords can appear in different inversions ($\overset{6}{5}$, $\overset{4}{3}$, and $\overset{2}{}$ chord in figured bass notation).

During the 19th century, chords with even more notes established. In tertian structure, the ninth interval (m9 or M9) above the root is the next to add. Composers of the later Romantic period occasionally use these ninth chords. In jazz harmony, the ninth and other additional **tensions** (9, 11, and 13, with alterations) play an important role.

¹⁶Usually, the distance between the bass and the lowest upper voice does not need to be minimal.



<i>Thirds</i>	(M3, m3, M3)	(M3, m3, m3)	(m3, M3, m3)	(m3, m3, M3)	(m3, m3, m3)
<i>Triad+7</i>	(M, M7)	(M, m7)	(m, m7)	([◦] , m7)	([◦] , [◦] 7)
<i>Name</i>	Major 7	Dominant 7	Minor 7	Half-diminished 7	Diminished 7
<i>Abbr.</i>	M ^{maj7}	M ⁷	m ⁷	◦7	◦7

Figure 2.13. Five seventh chord types used in Western classical music. We show the chords in root position above C4. The first row indicates the specific thirds for constructing the chords, the second row denotes the chords as a compound of triads and seventh interval above the root.

2.6.2 Nonchord Tones

Apart from the simultaneous appearance of chords (chorale style or **block chords**), composers make use of melodic elements to artistically shape harmonic constructs. We call the musical **texture** to be **homophonic** when mainly being constructed from block chords. As for the opposite observation, **polyphonic** music exhibits voices that are independent in rhythm and melody.

For chord-based concepts of music analysis, the homophonic texture is the default. Here, the rhythm of the music—marked by onsets of instruments or voices—coincides with the **harmonic rhythm** generated by the change of chords in the abstract sense. The umbrella term **figuration** summarizes all deviations from this homophonic structure. We speak of **rhythmic figuration** when repeating notes without any change in pitch. **Harmonic figuration** refers to chord notes sounding successively after each other—known as **broken chord** or **arpeggio**. In most situations, we perceive these structures as variations of chords rather than as melodic lines due to the strong completeness of chords. J. S. Bach’s famous Prelude in C major BWV 846 is an example for this psychoacoustical phenomenon.

All other melodic elements involve pitch classes outside the current chord denoted as **non-chord tones**. This **melodic figuration** makes use of additional notes to fill gaps between chord tones and to smooth the melodic lines of the voices. That way, they contribute to the horizontal aspect of harmony and touch the fields of voice leading and counterpoint. Usually, the nonchord tones are part of the underlying scale. Sometimes, chromatic alterations of scale notes appear as well. In Romantic harmony, notes from other scales often serve as nonchord tones.

There are different categories of nonchord tones depending on the way they are approached and left, and on their metrical position [69, 122, 191, 204]. In the following, we explain the different types by means of the example in Figure 2.14.

- **Passing tones** appear within a stepwise, unidirectional motion. There are accented passing tones—placed on a strong beat—or unaccented ones. In Figure 2.14, we find an unaccented passing tone in Measure 7, Beat 1+ (F[♯] in the bass). An accented passing tone occurs in Measure 9, Beat 2 in the bass (F[♯]) resolving to E as third note of the triad C[♯][◦].
- **Neighbor notes** depart stepwise from a chord tone and return. In Measure 1, Beat 3+, we find an unaccented neighbor note (D[♯] in the alto). A **neighbor group** comprises upper and lower neighbor notes within one motion.
- **Incomplete neighbor notes** arise when considering leapwise motion. The tenor F[♯] in Measure 1, Beat 3+ is an unaccented example. Particular types of incomplete neigh-

Jesu, meine Freude

Motette III

J.S. Bach (1685-1750)

BWV 227

1. Choral

Sopran
Je - su, mei - ne Freu - de, mei - nes Her - zens Wei - de,
ach wie lang, ach lan - ge ist dem Her - zen ban - ge,

Alt
Je - su, mei - ne Freu - de, mei - nes Her - zens Wei - de,
ach wie lang, ach lan - ge ist dem Her - zen ban - ge,

Tenor
Je - su, mei - ne Freu - de, mei - nes Her - zens Wei - de,
ach wie lang, ach lan - ge ist dem Her - zen ban - ge,

Baß
Je - su, mei - ne Freu - de, mei - nes Her - zens Wei - de,
ach wie lang, ach lan - ge ist dem Her - zen ban - ge,

5
Je - su, mei - ne Zier, Got - tes Lamm, mein Bräu - ti - gam,
und ver - langt nach dir!

Je - su, mei - ne Zier, Got - tes Lamm, mein Bräu - ti - gam,
und ver - langt nach dir!

Je - su, mei - ne Zier, Got - tes Lamm, mein Bräu - ti - gam,
und ver - langt nach dir!

Je - su, mei - ne Zier, Got - tes Lamm, mein Bräu - ti - gam,
und ver - langt nach dir!

9
au - ßer dir soll mir auf Er - den nichts sonst Lie - bers wer - den.
au - ßer dir soll mir auf Er - den nichts sonst Lie - bers wer - den.
au - ßer dir soll mir auf Er - den nichts sonst Lie - bers wer - den.
au - ßer dir soll mir auf Er - den nichts sonst Lie - bers wer - den.

Figure 2.14. Opening choral from J. S. Bach's motet "Jesu, meine Freude." We display the score in a public engraving by Alvarez using the free software Lilypond. The source file is available at <http://www.uma.es/victoria/varios.html>.

bors are the unaccented **escape tone**—coming from a stepwise motion and resolving by a leap in the opposite direction—and the **appoggiatura**—an accented neighbor approached by a leap and resolved by an opposite step.

- **Anticipations** are unaccented notes that become part of the following chord. We see an anticipation in Measure 10, Beat 2+ (soprano C♯).
- **Suspensions** are chord notes from the previous chord (prepared) and resolve downwards after the chord change—performing a rhythmic delay. For a suspension, the preparation of the tone in the same voice is essential. In Measure 5, Beat 3, we see a suspension over the chord BM (E resolving to D♯ in the tenor). This $4-3$ suspension (fourth to third above the bass note) and the joint $6-5$ / $4-3$ double suspension are the most frequent forms in Western classical music.¹⁷ Other types are $2-3$, $7-6$, and $9-8$ suspensions. Sometimes, the resolution of a suspension coincides with the next chord change. The analogue to the suspension in upward direction is called **retardation**.
- **Pedal points** are sustained notes while the other voices change chords. Most often, they constitute prolongations of the tonic note $\hat{1}$ or the dominant note $\hat{5}$.

The different manifestations of figuration can appear in various combinations and successions. Altogether, figurative elements constitute a crucial aspect of musical style.

2.6.3 Functional Harmony and Chord Progressions

As for pitch classes, the relation of chords to a reference note (or chord) accounts for their diatonic function. For this reason, a similar terminology became established—known as **functional harmony**. Rameau [197] first proposed ideas for such a system, which Riemann [200] elaborated. Later, Maler [145] contributed to a standardization of terms and symbols in the German tradition.

With terminology of functions, it is important not to confuse *notes* (tonic note) with *chords* (tonic chord—a triad built upon the tonic note). We therefore specify the tonal construct when referring to functional names. In functional harmony, we group diatonic functions into three main categories according to the principal chords on $\hat{1}$, $\hat{4}$, and $\hat{5}$ —tonic, dominant, and subdominant chord. The chords within a class are related as **parallel chords** ($M \xrightarrow{\text{down } m3} m$ and $m \xrightarrow{\text{up } m3} M$) or **contrast chords**¹⁸ ($M \xrightarrow{\text{up } M3} m$ and $m \xrightarrow{\text{down } M3} M$). In the tradition of functional theory, the diminished chord on scale degree $\hat{7}$ in major is regarded as an “incomplete” dominant seventh chord on $\hat{5}$ (missing root note). This interpretation is not in compliance with the historical evolution of this chord and leads to problems when interpreting chord progressions.¹⁹

This is one reason why we prefer a different analysis system—**Roman numeral analysis**—referring to chords as numbers [69, 219, 249]. For example, “V” refers to a major triad on the scale degree $\hat{5}$, “iii” denotes a minor triad on $\hat{3}$, and “♯iv^o” indicates a diminished triad on the altered scale degree $\hat{\sharp}4$. Sometimes, capital roman numerals also refer to the triads’ roots

¹⁷Note that these suspensions are *nonchord* tones and, thus, no *chords*. For this reason, it is not correct to speak of a “suspended *chord*” (“sus⁴”) in classical harmony. Similarly, the $\frac{6}{4}$ double suspension is no chord—even though it looks like a triad inversion.

¹⁸In German: Gegenklang, or Leittonwechselklang.

¹⁹In diatonic “circle of fifths” sequences, diminished triads appear as individual chords with the $\hat{7}$ acting as a root note. Another example are cadences, where voice leading rules do not indicate incompleteness of this chord. Gárdonyi and Nordhoff expose the problems of this “incomplete chord” concept [69, p. 15].

<i>Chord type</i>	M	m	m	M	M	m	°
<i>Function name</i>	Tonic	Subdominant parallel	Dominant parallel	Subdominant	Dominant	Tonic parallel	Incomplete dom. ⁷
<i>Function short</i>	T	Sp	Dp	S	D	Tp	\mathcal{D}^7
<i>Roman numeral</i>	I	ii	iii	IV	V	vi	vii°

m	°	M	m	M	M	M	°
Tonic	Incomplete dom. par. ⁷	Tonic parallel	Subdominant	Dominant	Subdominant parallel	Dominant parallel	Incomplete dom. ⁷
t	\mathcal{d}^7	tP	s	D	sP	dP	\mathcal{D}^7
i	ii°	III	iv	V	VI	VII	vii°

Figure 2.15. Scalar triads of the major and minor scales. The upper part shows the triads appearing in the major scale. The lower part displays the most important triads of the natural and harmonic minor scales. In the first row, we denote the triad type. The next two rows indicate the diatonic function according to Riemann and its abbreviation (lower-case letters refer to minor chords). In the last row, we mark the roman numerals for the chords. For the harmonic minor mode, the altered leading tone results in a major dominant chord V—just as for the major scale.

without further indicating the chord types. Figure 2.15 gives an overview of the different terminology for the major and the minor scale.

Besides the structure and function of chords—and their ornamental variation—, the way of connecting chords plays an important role for perception of tonality and musical style. Typical chord progressions appear frequently within musical styles (and across them). The most important motions are the following:

- A harmonic **pendulum** denotes the succession of a chord progression and its backward motion. Often, they appear with the tonic chord as frame chord and serve to establish or stabilize the key at the beginning of a section. Frequent samples are I-V-I and I-vii°-I in major or i-V-i and i-vii°-i in minor.
- **Sequences** are successions of root note progressions that repeat a pattern of one or more intervals. They can either stay in the pitch class content of the actual scale (diatonic sequences) or employ other scales while preserving the specific interval size of the progressions (real sequences). In general, sequences provide high harmonic motion—often in association with a fast harmonic rhythm. The most important example is the “circle of fifths” sequence consisting of concatenated descending fifth progressions: I-IV-vii°-iii-vi-ii-V-I (in major).
- **Cadences** are the ubiquitous ending sequences in Western harmony. They arise from combinations of the melodic “clausulae” in Early music. The most important cadences (in major) are ii-V-I, IV-V-I, and IV-vii°-I. As opposite to the **authentic cadence** V-I as a falling fifth progression ($M \xrightarrow{\text{down } P5} M$), the **plagal cadence** IV-I with a rising fifth ($M \xrightarrow{\text{up } P5} M$) is less common and sometimes dedicated to particular effects—such as the “A-men” in church music. Apart from this, a **half-cadence** or **imperfect**

Table 2.2. Categorization of root note progressions. Here, we display an overview of the authentic and plagal categories of root note progressions. Progressions by complementary intervals in opposite direction belong to the same category.

Interval	Δ	Complem.	Δ	Quality
P1	0	P8 \searrow	-12	None
m2 \nearrow	+1	M7 \searrow	-11	Authentic
M2 \nearrow	+2	m7 \searrow	-10	Authentic
m3 \nearrow	+3	M6 \searrow	-9	Plagal
M3 \nearrow	+4	m6 \searrow	-8	Plagal
P4 \nearrow	+5	P5 \searrow	-7	Authentic
+4 \nearrow	+6	$^{\circ}$ 5 \searrow	-6	None
P5 \nearrow	+7	P4 \searrow	-5	Plagal
m6 \nearrow	+8	M3 \searrow	-4	Authentic
M6 \nearrow	+9	m3 \searrow	-3	Authentic
m7 \nearrow	+10	M2 \searrow	-2	Plagal
M7 \nearrow	+11	m2 \searrow	-1	Plagal
P8 \nearrow	+12	P1	0	None

cadence ends on the dominant chord (V) and, thus, constitutes a rather weak feeling of arrival that calls for continuation (towards the tonic chord).

To categorize chord progressions, we extend the system of plagal and authentic cadences to all chord progressions—as proposed by Bárdos [14, 69]. Hereby, **authentic** progressions comprise root note movements of descending fifth and third intervals as well as ascending second ($\hat{=}$ descending seventh) interval progressions. **Plagal** progressions are of opposite direction (see Table 2.2). These qualities only refer to pitch classes and are independent from the octave of the notes. For that reason, progressions by complementary intervals in the opposite direction belong to the same category. The ratio between authentic and plagal chord progressions in music appears to be characteristic for a specific musical style [69].

2.7 Key and Modulation

Finally, we want to introduce the concept of musical key, which is essential for music from the common-practice period. Both chords and the scale are important for establishing a key [45]. There are different theories to explain their interdependency.²⁰ The theory of “Stufen” (scale degrees) departs from the scale as preexisting material and deduces the chords as triads on the scale degrees [219, 249]. As opposite to this, the theory of functions proceeds from the principal triads (tonic, dominant, subdominant) and derives the scale as the sum of these chords’ pitch classes [51, 200, 214]. Whereas the scale constitutes a pitch class set with a pronounced starting *note*, a key is defined by a referential major or minor *chord* that marks the center of gravity. For this subjective sense of arrival and rest, both the scale and particular chord progressions—such as cadences—play an important role but are not invariable. Human key perception shows a certain invariance against scale variations such as chromatic inflection of chords [45]. Examples are the Neapolitan sixth chord (as altered

²⁰Basically, we find the same controversy as the one between Riemann’s functional harmony and the Roman numeral analysis (Section 2.6.3).

subdominant) or the Picardy third (a major final chord at the end of minor key pieces—such as the last chord in Figure 2.14). In summary, we define a **key** as “a set of pitch relationships that establish a note—or, better, a chord—as a tonal center” [204, p. 43].

In major-minor tonality, we name the 24 keys after their corresponding tonic chord: G major is the key with the tonic chord GM. The G major scale provides the most important pitch classes for this key. This is indicated by the **key signature** (accidentals at the beginning of the staff such as the \sharp sign next to the clef in Figure 2.14). Nevertheless, other pitch classes arise as well—with particular harmonic purposes. Apart from the **global key**—often mentioned in the work title such as “Symphony in G major”—, parts of a movement may exhibit different **local keys**. These foreign key regions often occur in the middle section of a movement. When the harmonic structure prepares the arrival of the new key, we speak of a **modulation** [204]. There are different types:

- **Diatonic modulations** use a **diatonic pivot chord**, which has different functions in the previous and in the new key.
- A **chromatic modulation** takes place when a pitch class or chord from the previous key is chromatically altered in order to obtain a new role.
- **Enharmonic modulations** make use of the enharmonic equivalence of pitch classes or chords. By re-spelling pitches, an altered chord receives a new function in the upcoming key.

Even if single notes or chords play a particular role, modulations typically constitute a longer process [214].

In Section 2.5, we saw that fifth-related scales share a high amount of pitch classes (Figure 2.6). Due to the close connection between key and scale, we can apply the concept of fifth measurement to keys as well. Closely related keys have a small fifth distance ($|\mathcal{D}| \leq 1$). The **circle of fifths** (Figure 2.16) visualizes these key distances [204, p. 466 ff.]. There are particular names for some key relationships:

- **Relative keys** share the same key signature and diatonic scale ($\mathcal{D} = 0$), for instance:

$$\text{F major} \xrightarrow{\text{down m3}} \text{D minor} \quad (2.30)$$

and vice versa.²¹ For pieces with a minor global key, the modulation to the relative major key is very common.

- **Parallel keys** share the tonic note but not the tonic triad ($|\mathcal{D}| = 3$):

$$\text{F major} \xrightarrow{\text{P1}} \text{F minor}. \quad (2.31)$$

- **Fifth-related keys** differ in one scale note ($|\mathcal{D}| = 1$), such as:

$$\text{F major} \xrightarrow{\text{up P5}} \text{C major}, \quad (2.32)$$

²¹Note the different traditions: In German, “Paralleltonart” denotes the relative key. The analogous chord relationship influenced the names of diatonic functions such as “tonic parallel” (compare Figure 2.15). The German equivalent for “parallel key” is “Varianttonart.”

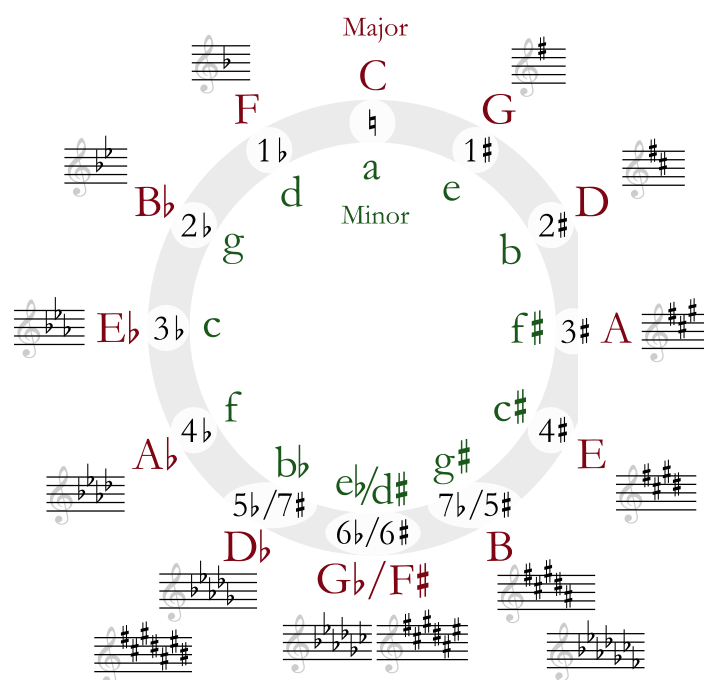


Figure 2.16. Circle of fifths for musical keys. Parallel major and minor keys share the same key signature; fifth-related keys are next to each other. For key signatures with more than five accidentals, the enharmonically equivalent key is shown as well (image from [261]).

but also the relatives with equal diatonic scale:

$$\text{F major} \xrightarrow{\text{up } M3} \text{A minor}. \quad (2.33)$$

Fifth-related keys are neighbors in the circle of fifths (Figure 2.16). The modulation to the upper fifth key is the most frequent one in pieces with a major global key.

- **Mediant keys** relate by a third interval between their tonic notes. Relative and parallel keys constitute mediant relations, but we also find modulations to chromatic mediants—especially in the Romantic period. An example is ($\mathcal{D} = +3$)

$$\text{F major} \xrightarrow{\text{down } m3} \text{D major}. \quad (2.34)$$

We avoid the problematic terminology of using functional names for keys. For example, the *dominant chord* has a specific tension towards the tonal center (tonic) and, thus, is always a major chord with a leading tone—also in minor keys. It is not helpful to speak of a “dominant key”, which—being the result of a modulation—constitutes a new tonal center itself with a new dominant chord. We therefore prefer the more neutral relative fifth measurement introduced in Section 2.5. In this notation, a “+1 key” can be a minor key as well since it has no dominant function.

2.8 Models of Musical Pitch

2.8.1 Consonance and Dissonance

In the previous sections, we saw several theories for explaining relationships between chords, scales, and other structures. Now, we also want to mention theories approaching pitch relations in a more fundamental way. It is important to consider the type of tonal structures these theories apply to.

One of the oldest classification models for pitch relations is the concept of **consonance** and **dissonance**. In Early music, most principles of counterpoint rely on the resolution of dissonant intervals compliant to specific rules. Nevertheless, the detailed categorization of intervals changed over time, constituting an important style indicator. From today’s point of view, consonance is “only a matter of degree, not of kind” [215, p. 21]. The common nomenclature of consonant intervals hints at that fact: **Perfect consonances** encompass the perfect intervals²²

$$P1, P4, P5, P8. \quad (2.35)$$

Major and minor thirds and sixths are called **imperfect consonances**:

$$m3, M3, m6, M6. \quad (2.36)$$

Because of this “imperfectness,” the final chords in Medieval and Early Renaissance music do not exhibit triad thirds. All other intervals constitute **dissonances**. In most tonal systems, they resolve following particular rules such as $2-3$, $4-3$, $6-5$, $7-6$, or $9-8$ (compare Section 2.6.2). One may specify degrees of dissonances as well—a major second interval sounds less harsh than a minor second. Summarizing these fine distinctions, the concept of consonance may relate to the location of pitch classes in the harmonic series of the reference tone. According to Schönberg, dissonances merely constitute “more distant overtones” [215, p. 45].

With the beginning 20th century, the novel handling of dissonances was the crucial step towards new tonal systems. This “emancipation of the dissonance” [216, p. 104] leads to a similar treatment of consonances and dissonances, no longer forcing a resolution of the latter. Musical pieces following such rules rely on the equal-tempered chromatic scale rather than on diatonic scales. For analyzing such pieces, theorists proposed particular systems such as the **pitch class set theory** [64, 86]. Here, we only consider unordered pitch class sets sounding either successively or simultaneously—also called **sonorities**. In Section 2.5, we already introduced the notation for this concept (Equation (2.22)). For instance, we can write an augmented triad as

$$S_+ = \{0, 4, 8\}. \quad (2.37)$$

For pitch classes, only six different interval types occur when ignoring the octave and unison. We can therefore order all possible pitch class sets into six **interval categories** (IC) by iterating the basic intervals [94, 196]. Table 2.3 lists prototypes for these categories with the pitch classes in ascending order. We apply suitable transpositions in order to start with $q = 0$.

²²Note that the perfect fourth behaves in a particular way. A fourth in relation to the root note—such as a $4-3$ suspension—constitutes a dissonance. In contrast, a fourth in a different context is consonant.

Table 2.3. Interval categories and prototypes of pitch class sets. The sets are constructed by iterating the interval distance Δ (mod 12). When the iteration reaches an already existing pitch class, the procedure starts again a semitone higher. Finally, we transform the sets to so-called “prime forms” by suitably transposing and inverting (table from [94]).

<i>Category</i>	Δ	<i>Prototypes</i>
IC1	1	{0, 1}, {0, 1, 2}, {0, 1, 2, 3}, ...
IC2	2	{0, 2}, {0, 2, 4}, {0, 2, 4, 6}, ...
IC3	3	{0, 3}, {0, 3, 6}, {0, 3, 6, 9}, ...
IC4	4	{0, 4}, {0, 4, 8}, {0, 1, 4, 8}, ...
IC5	5	{0, 5}, {0, 2, 7}, {0, 2, 5, 7}, ...
IC6	6	{0, 6}, {0, 1, 6}, {0, 1, 6, 7}, ...

2.8.2 Geometric Pitch Models

Beyond the presented concepts, there are theories that try to explain pitch relations by means of geometric models. They usually “correlate spatial distance with intuitive musical distance.” [135, p. 42]. Originating from tuning theories, they served to rapidly calculate frequency relations. We already introduced several geometric models such as Shepard’s pitch helix (Figure 2.2) or the circle of fifths (Figure 2.16), which sometimes also applies to pitch classes. Another historical concept is Weber’s regional chart [249]. Euler’s “Tonnetz”—primarily developed for representing just intonation—inspired the theories of Riemann and Cohn [41]. A spatial visualization of the Tonnetz results in a toroidal structure [13]. All of these concepts give major importance to perfect fifth relations. Moreover, they consider major and minor third axes that are important for chords and keys (relative, parallel, and other mediant relations).

More recent models have refined these ideas to better account for the different perception of pitches, chords, and keys. They also take into account the results of psychoacoustic studies such as the ones by Krumhansl [124]. Gatzsche and Mehnert [71, 72] proposed a symmetry-based model that separately considers key-related (diatonic) and key-spanning (chromatic) properties. Chew [34, 35] developed a model named “spiral array” with a special emphasis on the determination of tonal centers. Lerdahl’s tonal pitch space [135] introduces several spatial models for pitch classes, chords, and keys. These levels interrelate by tree-like structures.

Theorists from the Hungarian tradition also consider symmetrical divisions of the octave. They particularly analyze the symmetries of scales that constitute the basic pitch material for sections of music [69, 134]. Some of these ideas are known as “Theorie der Tonfelder” [82].

Most of the mentioned theories employ complex and high-dimensional models to explain tonal relations. Often, these models serve to explain particular musical structures, styles, or even single composers’ techniques. Sometimes, a clear discrimination of the concerned types of tonal structures is missing. In this thesis, we do not use complex spatial models. Rather, we attempt to understand how the general types of tonal structures are responsible for musical style. However, we make extensive use of relations by perfect fifths as the most basic pitch class relation. This may be a justified assumption when dealing with Western classical music since “only the fifth cycle is basic to the diatonic system, which in many respects is asymmetrical” [135, p. 45].

2.9 Tonal Complexity

Beyond the concrete treatment of specific tonal structures such as intervals or chords, more abstract concepts are useful to describe the overall nature of tonality. Theorists proposed different notions for such purpose. One idea is a “degree of tonality” [92] in the specific sense of “keyness” [100] or “keystrength” [124]. Another idea in the literature is the definition of “tonal tension” [135,136]. We summarize such concepts under the term **tonal complexity**. Relying on the introductory parts of [257], we discuss the characteristics of this notion respecting the hierarchical nature of tonality in the time domain. In Section 6.2, we compile a set of concrete musical assumptions for a quantitative measure of tonal complexity. Based on these hypotheses, we design experiments for testing our proposed tonal complexity measures regarding different temporal scales.

In Western art music, one major purpose of harmony is to emphasize musical structure. Typical harmonic phenomena serve to highlight pivotal moments of a composition. This observation applies to different time scales. Local structures such as intervals or chords show different characteristics with respect to harmonic stability, creating a feeling of either tension or resolution. Progressions of these items over time such as pendula, sequences, and cadences form larger lines of development by employing chords of appropriate quality. Over the course of a work, the structural parts may differ significantly with respect to their tonal characteristics. A section that is harmonically stable may be followed by a contrasting section that feels rather unstable or tense. These contrasts serve to create the arc of tension of a musical piece. In the sonata form, for example, the unstable development part stands between the more stable exposition and recapitulation phases.

Apart from such intra-work aspects, there is a related but more abstract quality describing the harmony of complete pieces or even a compositional style. The pitch class selection of Western music evolved from a diatonic scale to a fully chromatic set of equally relevant pitches in the atonal period [190]. The applied chords and chord progressions became more complex—on a rough scale—over the centuries. We find a similar behavior for the complexity with respect to larger formal structures. For example, the number and harmonic distance of modulations in Romantic pieces is usually much higher than for Classical works. LaRue [130] described such kind of tension as one of the basic functions of harmony and discusses the stylistic impacts of such phenomena.

For all these different aspects of tonality, pitch class distributions may constitute a useful source of information. Regarding local tonal structures such as chords, the quality of pitch class sets and their characteristic intervals is crucial (compare Section 2.8.1). For coarser time scales up to a complete movement, pitch class histograms may provide information about tonal complexity since their flatness relates to the amount and the type of modulations and the relationship of local keys. Motivated by this, we propose in Section 6.2 several measures based on pitch class representations and test their behavior with respect to several musical assumptions.

2.10 Tonality Aspects of Musical Style

We mentioned the interaction between musical style and the use of certain tonal elements several times. As a concluding remark of this chapter, we want to summarize these ideas and discuss the overall impact of tonality for style recognition. Parts of this discourse follow the introduction of [256].

When addressing Western classical music, musicologists often prefer the detailed view. They find a great individuality in the style of single composers, together with substantial evolutions and breaks within their oeuvre. These subtle stylistic differences may arise “partly because of the differing attitudes of societies and composers” [182]. The balance between a composer’s **personal style** and a time-related **contemporary style** or **epochal style** changed over the course of music history [182]. Out of many theorists who discussed this relation, we point to de la Motte [51, 52] who linked the debate with harmony analysis.

In any case, one can observe lines of development in music history as well as the breaking of such lines. Because of that, many researchers and listeners divide the repertoire of Western classical music into **historical periods** or **eras**. Such a categorization inevitably constitutes a simplification but can provide “a reasonably consistent basis for discussion” [74]. Treating such task with success provides a starting point for analysis and may precede a closer look at individual stylistic tendencies [65, 250].

Some researchers illustrate the homogeneity of periods with a “unique artistic and intellectual spirit” and focus on each periods’ new achievements [240]. Others treat the style of a specific era and its inner coherence [28, 205]. Clarke [40] makes attempts towards a more detailed view by taking into account different sub-phases of eras. He claims styles to begin in an experimental phase, to grow to an established language, and to die after an elaborate ending period. Beyond the historical context, style classes often relate to geographical categories and may exhibit influences of local folk culture or particular social conditions. Adler [3] determines three types of style definition relating to *time*, *place*, and *author*. He estimates the time-related categorization as the “essence of independent style-criticism” but on the other hand values author identification as “style-criticism in its highest form” that, however, “sometimes turns on subordinate details.”

Looking at a piece of music, we further have to devote attention to the specific musical genre²³ and the possibilities of the instruments. The refined distinctions between style and idea, genre, or form are of major importance. The choice of a **genre** determines the external conditions; a genre usually exists throughout different periods but may play a more important part in one of those periods. The **idea** is the primary factor of a concrete piece, its individual element. Often, the idea relates to the melodic domain but also elements concerning other parameters may serve as musical idea. **Form** is the shape or structure of a piece with respect to time, thus dealing with aspects such as repetition, variation, and development.

According to Belaiev [19], a composition is “the result of giving form to an idea.” **Style** is one of the factors *how* to do this. As complementary notions, style and idea may embody “the general” versus “the particular” [182]. In comparison, most scholars consider the idea as a work’s more important and prominent constituent [19, 216]. This is one challenge for style analysis: style constitutes a deeper layer, often covered by the idea and external requirements. Some researchers propose to depart from an analysis of form followed by the detailed analysis of content [3]. Others stress the importance of the details—in relation to the whole—and claim statistical analysis of certain style indicators to be an appropriate method [206].

Concerning such **style indicators**, harmony constitutes one domain—besides sound, form, rhythm, and melody [129]. The situation is complex because of a high interdependency of these categories. Their relationship itself changes over history. Apart from the sound with its “psychological firstness” [129], many researchers ascribe high importance to tonality and

²³Here, the term “genre” (German “Gattung”) denotes a particular type of work, usually connected to a defined instrumentation, a musical form model, and sometimes with an external purpose. Examples are the mass, the opera, the piano sonata, the string quartet, or the symphony.

notice “clear conventions of harmonic behavior” within an era [130]. Belaiev [19] stresses the importance of “chordal combinations” and harmonies in general for defining a style. For Rosen [205], the establishment of a new style refers to all musical parameters in a way “that all the contemporary elements of musical style [...] work coherently together.” Nevertheless, he emphasizes “the musical language [...] of tonality” as an essential precondition for the classical style. As a musical dimension, harmony is widely independent from timbral properties such as instrumentation, playing techniques, or singing style. Therefore, we may find important aspects of the deeper layer “style” in a work’s harmonic characteristics.

In his overview article [129], LaRue proposed a list (“sample outline”) of stylistic properties as a guideline for style analysis. With respect to harmony, he ascribes importance to—among others—the following characteristics:

- **Large-scale tonal relationships, key-schemes, harmonic motifs.** This broad dimension has particular meaning before concentrating on details. Here, the global key, secondary keys, and key relationships to other movements play a role.
- **Modality, chromaticism, polytonality.** These properties mainly relate to the pitch class content and scales in use. Gárdonyi and Nordhoff [69] announced various observations in this field.
- **Chord vocabulary, alterations, dissonances, progressions, modulations, harmonic rhythm.** Here, not only modulations to various keys, but also their relative emphasis (length, weight) matter. De la Motte [52] also remarked the meaning of the chord vocabulary and its historical evolution. Others emphasize the use of specific chord progressions and modulation routes [14, 69]. LaRue rates the treatment of dissonance and chromaticism as crucial for a composer’s individuality.
- **Imitation, voice leading, texture, counterpoint.** These details of part writing and the general interrelation of voices constitute a central stylistic aspect of some historical periods.
- **Text influence, affective chords.** Though this category is only relevant for text-based music, it is of high importance for style analysis. With respect to harmony, a single chord or key may suffice for expressing a mood.

In his later book [130], LaRue further abstracts beyond the conventions of the common-practice period. He defines **color** and **tension** as the most basic functions of harmony that are not to confuse. These functions exist on various time scales.

In this section, we pointed out the difficulty of defining and analyzing the abstract notion of style in music. Style properties may hide behind many other—and, more predominant—characteristics of a piece. Nevertheless, we may always find stylistic peculiarities in a musical work—and there are good reasons to look for them in the field of harmony and tonality.

3 Technical Foundations

Humans produce music in order to be perceived by other humans or by themselves. Therefore, we may regard music as a form of communication or artistic expression. Physically, musical sounds—as all sounds—are fluctuations of the local air pressure level, which propagate to the listener’s ears as longitudinal waves. Researchers [10, 160, 171] as well as composers [244, 263] led an intensive debate about how to define music and where to draw the separation line between music and non-musical sounds. Today, there is no agreement about that. Nevertheless, it is clear that there are several types of music that do neither exhibit harmonic sounds (tones) nor clear metrical structures.

The most common form of music experience is the human **performance** with people playing in front of an audience. For more than hundred years, technical methods exist to store the acoustic impression of performances in the form of **music recordings**. Section 3.2 outlines the technical properties of such audio recordings.

When talking about a **musical work** or **composition**, we assume that this specific piece of music is reproducible. For music from the common-practice period, the traditional form of transmitting and preserving music is the **musical score**. Apart from such written documents, technical advancements of the last decades enabled further ways of storing the parameters and instructions for human or automatic music performances. In Section 3.1, we will present several kinds of such **symbolic music representations**.

This dissertation deals with automatic methods for analyzing audio recordings. The first step in most systems is the extraction of suitable **features** for describing properties of the audio data. In Section 3.4, we show several common feature types that mostly relate to the timbre of the music. Some of these features rely on spectrograms, which we introduce in Section 3.3. Section 3.5 presents features that describe the tonal content of the music on a low and intermediate semantic level.

3.1 Score Representations and Symbolic Data Types

In many cultures, people transmitted musical pieces by means of oral tradition. Through the history of Western art music, the use of written documents that indicate clues for the performance of pieces obtained more and more importance. In ancient and medieval times, signs served to roughly indicate pitch change direction—the **neumes**. Later, the Roman **square notation** introduced first note symbols of today’s kind. Over the centuries, the **five-line staff** established and an increasing number of symbols served to determine more and more musical parameters such as articulation, dynamics, and expression [227].

The most detailed type of notation is the **full score**, which provides a separate staff for every instrumental or vocal part, or for small groups of such parts. Figure 3.1 shows the first score page of L. van Beethoven’s “Fidelio” overture for full orchestra. For notation of common-practice orchestral music, the traditional order is—from top to bottom—woodwind instruments, brass instruments, percussion instruments, soloists or choir, and string instruments. For historical and practical reasons, the notation of some wind instruments makes use

Ouvertüre zu Fidelio

Ludwig van Beethoven (1770-1827)
Op. 72c

The musical score is presented in a standard orchestral layout. It begins with the tempo marking 'Allegro' and the dynamic 'f'. The woodwinds (Flutes, Oboes, Clarinets, Bassoons) and strings (Violins, Violas, Cellos, Double Basses) play a rhythmic, ascending motif. The brass instruments (Trumpets, Trombones) provide harmonic support. The score transitions to 'Adagio' with a 'p dolce' marking, where the strings play a more sustained, melodic line. The woodwinds and brass are mostly silent during this section.

Figure 3.1. Overture from L. van Beethoven’s opera “Fidelio” op. 72c. We display the first page in a music engraving by Oram using Lilypond. The score and the source files are available under creative commons public domain license at the homepage of the Mutopia project <http://www.mutopiaproject.org>.

Figure 3.2. Piano reduction of the “Fidelio” score page. The pitches comprise the most important components from the full orchestral version as shown in Figure 3.1. The text marks roughly indicate the instrumentation of the music: “G. Orch.” stands for the full orchestra (“Großes Orchester”) and “Hrn.” for the french horn section.

of a transposition. In Figure 3.1, for example, the french horns (“Corni in E”) are sounding a minor sixth interval lower than indicated by the notes.

Full scores are the most important source for accomplishing a musical performance since they contain the most detailed musical information as provided by the composers themselves. From the full score, the conductor gets the overview of all parts that the individual instruments are playing. Beyond that, more compact representations of the essential musical content¹ are useful for several purposes. When compressing a full score to a piano system—a pair of staves, often with treble and bass clef jointly—we speak of a **piano reduction** or **piano score**. Répétiteurs use such piano versions (“vocal scores”) to rehearse with singers; pianists also artistically perform piano transcriptions of orchestral works—sometimes arranged for two or more pianos. Figure 3.2 shows a piano reduction of the first “Fidelio” score page (Figure 3.1). The piano reduction does not necessarily contain all the pitches from the original score in order to be readable and playable on a piano. In our example, some of the timpanis’ and trumpets’ pitches are missing due to musical reasons.

Traditionally, scores are hand-written or printed on paper. For accessing scores with computers, it is common to convert printed sheet music into digital images using scanners. Such type of graphical score data is publicly available on a number of web pages such as the International Music Score Library Project (IMSLP).² For enabling computers to read the musical information from scores, we need a different data format with an explicit encoding of musical information [162]. Examples for such **symbolic representations** of music are the commercially developed MusicXML format [79] or a related type created by the open-source project Music Encoding Initiative (MEI).³ We may consider the source code of the engraving software Lilypond⁴ as another symbolic representation. In Figure 3.3, we show the MusicXML encoding of the Violin I part (first measure) from the Beethoven score.

A further symbolic format widely used by musicians is the MIDI [99] format (Musical Instrument Digital Interface), a technical standard protocol originally developed for the intercommunication of electronic instruments. A MIDI file consists of several event messages that are specified through a set of parameters such as *pitch*, *volume*, *key velocity*, or *channel number* (“note on” event). With a corresponding “note off” event, we can derive the duration of a note. The MIDI pitch number range is $p \in [0 : 127]$ with $p = 69$ corresponding to the

¹In Western music, this most often refers to the main melody, the bass, and an excerpt of the harmonic accompaniment (the basic chords).

²<http://www.imslp.org>

³<http://www.music-encoding.org>

⁴<http://www.lilypond.org>

<pre> <?xml version="1.0" encoding="UTF-8"> <movement-title>Ouvertüre zu Fidelio</movement-title> <identification> <creator type="composer">Ludwig van Beethoven (1770-1827) Op. 72c</creator> </identification> <part-list> <score-part id="P1"> <part-name>Violine I</part-name> </score-part> </part-list> <!-------> <part id="P1"> <measure number="1"> <attributes> <divisions>2</divisions> <key> <fifths>4</fifths> <mode>major</mode> </key> <time symbol="cut"> <beats>2</beats> <beat-type>2</beat-type> </time> <clef> <sign>G</sign> <line>2</line> </clef> </attributes> <direction placement="below"> <direction-type> <dynamics> <f/> </dynamics> </direction-type> </direction> </pre>	<pre> <note default-x="145"> <pitch> <step>E</step> <octave>5</octave> </pitch> <duration>3</duration> <type>quarter</type> <dot/> <stem default-y="-40">down</stem> </note> <note default-x="210"> <pitch> <step>B</step> <octave>4</octave> </pitch> <duration>1</duration> <type>eighth</type> <stem default-y="-55">down</stem> </note> <note default-x="240"> <pitch> <step>B</step> <octave>4</octave> </pitch> <duration>2</duration> <type>quarter</type> <stem default-y="-55">down</stem> <notations> <articulations> <staccato default-x="4" default-y="-7"/> </articulations> </notations> </note> <note default-x="300"> <rest/> <duration>2</duration> <type>quarter</type> </note> </measure> </part> <!-------> </score-partwise> </pre>
--	--

Figure 3.3. MusicXML encoding of the Violin I part from Beethoven’s “Fidelio” overture. We show the MusicXML commands for Measure 1 (Figure 3.1, staff 11). The first blocks (left hand side) refer to the preamble and the definition of key and time elements followed by the encoding of the dynamics indication (“forte”). The right hand side contains the encoding of the four note elements with the last one being a rest.

concert pitch A4. We already introduced this notation in Equation (2.10). Apart from an event list, we can also graphically display the pitch and time information from a MIDI file. Figure 3.4 shows such type of **piano roll** representation.⁵

On the web, we find large collections of symbolic music data that is publicly available. The most established data type is the MIDI format. Hereby, the quality range of the data is wide and we can find a lot of MIDI files with considerable errors compared to the pieces’ scores. Furthermore, different types of MIDI files exist. Since the MIDI data relies on a

⁵This term comes from the early automatic pianos, which used a roll of paper with holes for mechanically encoding a performance.

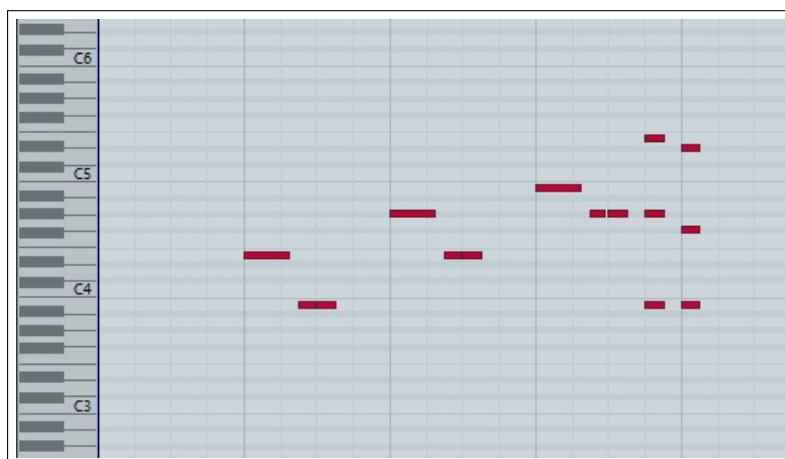


Figure 3.4. Piano roll representation of a MIDI file We visualize the first four measures of the Violin I part from Beethoven’s “Fidelio” overture by displaying the MIDI events as a piano roll. The bars indicate note events of a specified duration.

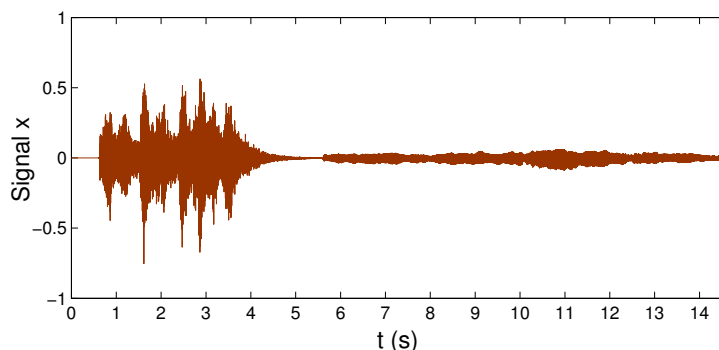
physical time axis and not on a musical one, the events can principally occur at every time. Automatically generated MIDI files—for example, exported from music engraving software—usually have a constant tempo, which can be different for individual sections. Beyond that, MIDI files can originate from human performances on respective instruments such as, for example, pianos with MIDI capabilities. These MIDI files have an additional information of human interpretation since they reflect the small tempo and rhythm deviations as made by humans (“performed MIDI”). Sometimes, this constitutes a challenge when trying to convert these MIDI files into scores. Beyond this, some musical information that is essential for Western music notation is missing in MIDI representations. Among others, this concerns the key signature or the enharmonic pitch spelling.

3.2 Audio Representations

For the human listener, music is an acoustic experience. A real performance of music by means of instruments or human voices contains much more information than we encode by means of a symbolic representation. For example, a listener may recognize an individual singer’s voice due to the specific timbre of his or her voice. Further aspects such as room acoustics or the relative positioning of the musicians affect the characteristics of a performance. The first methods for recording performances stored the acoustic signal in an analog fashion. Examples are phonograph records or magnetic tapes. An analog music recording constitutes a real-valued **continuous-time signal**. By the end of the 20th century, digital technologies found their way into the field of audio applications. The compact disc (CD) became the first publicly used medium for storing music in a digital representation. Such representations describe the audio content as a finite amount of numbers.

For converting analog signals into the digital domain—a process called **digitization** or analog-to-digital (AD) conversion—, two steps are necessary. First, we transfer the continuous time axis into a discrete set of time instances, which is known as **sampling**. We obtain a **discrete-time signal** that we may regard as a function $x : \mathbb{Z} \rightarrow \mathbb{R}$. The most common method is **equidistant sampling**. Hereby, we take the samples $x(n)$ from the analog

a) Orchestra Recording



b) Piano Recording

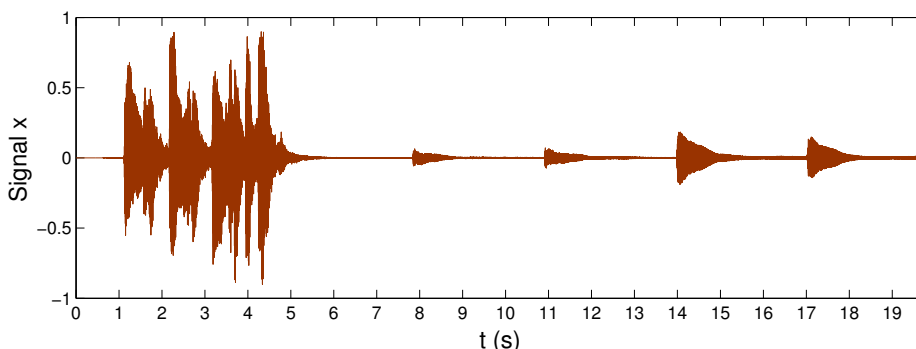


Figure 3.5. Waveforms of two audio recordings of Beethoven’s “Fidelio” overture (Measures 1–8). The first version (a) is a recording by the Slovak Philharmonic Orchestra conducted by Stephen Gunzenhauser (1988). The second example (b) is an arrangement for piano four hands by Alexander Zemlinsky. The pianists are Dennis Russell Davies and Maki Namekawa (2008). Instead of the sample numbers n , we indicate the corresponding physical time.

recording at equally spaced time points $t = n \cdot T \in \mathbb{R}_0^+$ with

$$n \in [0 : N - 1] := \{0, 1, \dots, N - 1\} \subset \mathbb{Z} \quad (3.1)$$

where $N \in \mathbb{Z}$ denotes the total number of samples [162]. The constant **sampling period** $T \in \mathbb{R}^+$ is the physical time distance between two neighboring samples. We express the number of samples per second with the **sampling rate** or sampling frequency

$$f_s := \frac{1}{T}, \quad (3.2)$$

usually given in Hertz (Hz). According to the Nyquist-Shannon sampling theorem, a digital signal with a sampling rate f_s allows for perfect reconstruction as long as the original signal has only frequencies up to the **Nyquist frequency** $f_s/2$. A CD recording typically has a sampling rate of $f_s = 44.1$ kHz and, thus, comprises the human hearing range reaching up to 20 kHz. For further details of the sampling procedure, we refer to [270].

As the second step, we represent the signal amplitudes $x(n) \in \mathbb{R}$ using a finite number of bits (**quantization**). An example is a uniform quantizer with a constant step size. Commercial audio CDs have a precision of 16 bits encompassing a range of $2^{16} = 65\,536$ amplitude values. For the details of quantization, we refer to the literature [162, 270].

The process of digitization described above is also known as **Pulse-Code Modulation** (PCM). The graphical visualization of an audio signal’s amplitude is called **waveform**. In Figure 3.5, we display the waveforms for two recordings of the first eight measures from “Fidelio,” one being an orchestra recording and the other one a piano transcription for four hands. The digital representation of both recordings is at CD quality with $f_s = 44.1$ kHz and 16 bit quantization.⁶ Looking at the waveforms, we first observe the different length of the signals—although they represent the same excerpt of the score. This results from a different tempo shaping of the performances. The Allegro motif in *forte* (Measures 1 – 4) has a comparable length of roughly 5 s in both recordings whereas the Adagio is slower in the piano version. In the *forte* part, the peak amplitudes reach higher values in the piano recording. Looking at the Adagio’s whole notes, we observe the difference between the decaying piano notes (Example (b), 8 s ff.) and the sustained horn notes in the orchestra version (Example (a), 5 s ff.)

3.3 Spectrograms

For understanding the physical and perceptual properties of audio signals, it turned out useful to analyze the signal’s frequency content. Hereby, we regard a signal as a mixture of sinusoidal components with different frequencies.⁷ The set of frequency coefficients regarding the individual sinusoids is called **spectrum**. To obtain the coefficients of a discrete-time signal $x : [0 : N - 1] \rightarrow \mathbb{R}$, we can compute the **Discrete Fourier Transform** (DFT) of size N , which is a complex-valued function $DFT_N : \mathbb{R}^N \rightarrow \mathbb{C}^N$. We obtain the **Fourier coefficients** $X(k) \in \mathbb{C}$ via

$$X(k) := \sum_{n=0}^{N-1} x(n) \exp\left(\frac{-2\pi i k n}{N}\right) \quad (3.3)$$

with $k \in [0 : K - 1]$ denoting the discrete frequency parameters ($K = N$).⁸ We obtain the physical frequency (in Hz) related to k by calculating

$$f_{\text{coeff}}(k) := \frac{k}{N} \cdot f_s. \quad (3.4)$$

To represent the discrete signal as a series with coefficients $X(k)$, we use the inverse DFT:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \exp\left(\frac{2\pi i k n}{N}\right) \quad (3.5)$$

The computation of all Fourier coefficients $X(k)$ requires $\mathcal{O}(N^2)$ operations, which may take a long time for large values of N . To speed up this process, we make use of the **Fast Fourier Transform** (FFT) algorithm [43]. This method recursively computes the DFT

⁶For most experiments in this thesis, we used audio recordings stored in compressed formats such as the MP3 format. Before applying further processing steps, we decode this data to a PCM audio representation and ignore the effects of possible audio coding artifacts. This may be justified since we use bitrates of at least 192 kilobits per second.

⁷Mathematically, any set of periodic functions may serve as basis function instead of sinusoids. However, sinusoidal functions turned out most convenient for computation.

⁸For real-valued signals, only the frequency parameters up to $K/2$ corresponding to the Nyquist frequency are relevant.

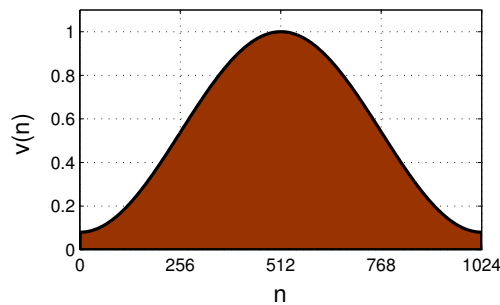


Figure 3.6. Hamming window function. Here, we display a Hamming window with a blocksize of $B = 1024$ samples.

by exploiting redundancies between the coefficients and, thus, reduces the computational complexity to $\mathcal{O}(N \cdot \log_2 N)$ operations. The recursion works particularly efficient if N is a power of two.

The DFT provides the frequency information of the whole signal x . The phase of the complex Fourier coefficients $X(k) \in \mathbb{C}$ encodes the time information with respect to the sinusoids. For analyzing the frequency content over time, we can use a local variant of the DFT called **Short-Time Fourier Transform** (STFT). To this end, we segment the signal into several windows or **frames** and estimate the sinusoidal components for each frame individually [68].

For the windowing procedure, we employ a discrete function $v : [0 : B - 1] \rightarrow \mathbb{R}$ with a length—or **blocksize**—of $B \in \mathbb{N}$ samples. The choice of this function is of major importance since the STFT describes the properties not only of the signal but also of the window function. It turned out beneficial to use bell-shaped windows such as the **Hamming** window (see Figure 3.6). We shift this window along the signal by a given amount of samples called **hopsiz**e $H \in \mathbb{N}$ (compare also Figure 5.7).

Applying the DFT to each of the resultings frames, we obtain the **discrete STFT**

$$\mathcal{X}(k, m) := \sum_{n=0}^{B-1} x(n + mH)v(n) \exp\left(\frac{-2\pi i k n}{B}\right). \quad (3.6)$$

Hereby, $m \in [0 : M - 1]$ denotes the frame index with the total number of frames

$$M \approx N/H. \quad (3.7)$$

In the matrix \mathcal{X} , each column $\mathcal{X}(\cdot, m)$ forms a **spectral vector** indicating the frequency content of the m -th frame. With a suitable hopsiz

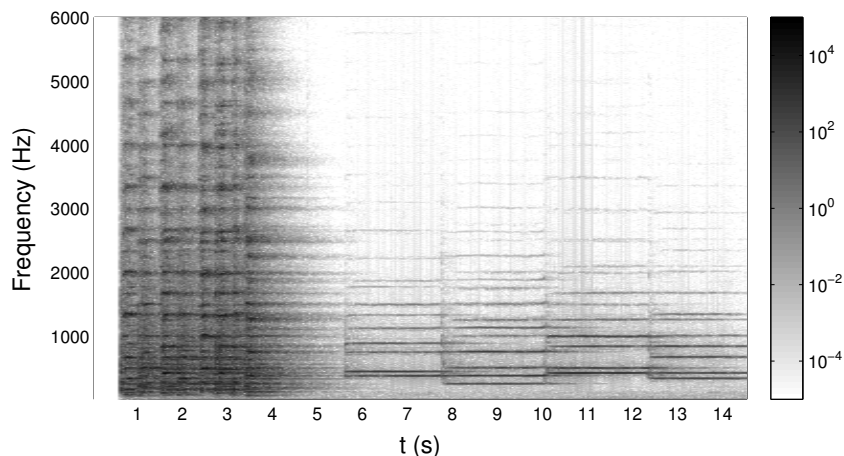
e H , the position of the frames gives a sufficiently fine time spacing to locate the frequency contributions for some applications. In this case, we can ignore the complex phase information,⁹ which leads us to the concept of a **spectrogram**¹⁰ \mathcal{S} :

$$\mathcal{S}(k, m) := |\mathcal{X}(k, m)|^2. \quad (3.8)$$

⁹In contrast, phase information is crucial in scenarios where a reconstruction of the signal should be possible such as, for example, in source separation applications.

¹⁰Sometimes, authors refer to $|\mathcal{X}(k, m)|$ as magnitude spectrogram and denote $|\mathcal{X}(k, m)|^2$ as **power spectrogram**.

a) Orchestra Recording



b) Piano Recording

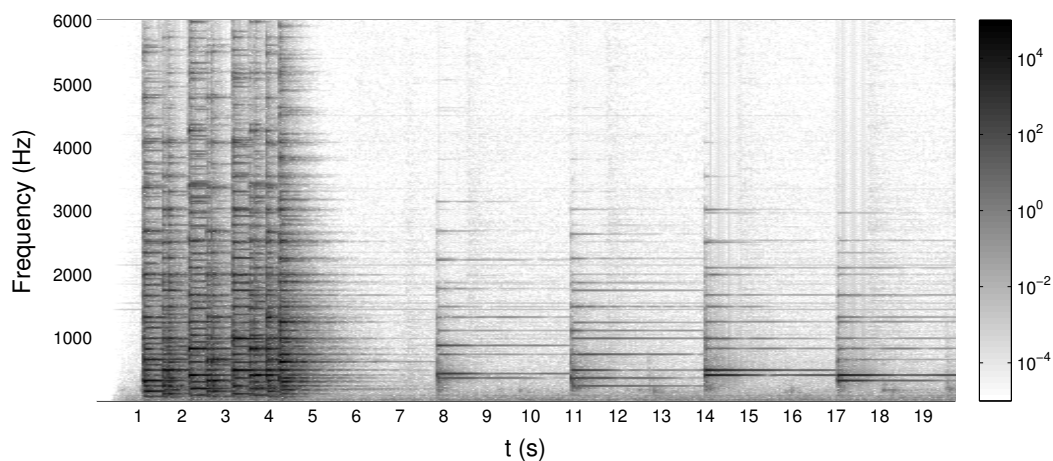


Figure 3.7. Magnitude spectrograms of the two “Fidelio” audio recordings. Here, we only display the frequency content up to $f = 6000$ Hz. In these time-frequency plots, we encode the amplitudes $\mathcal{S}(k, m)$ by means of different gray levels.

One entry $\mathcal{S}(k, m)$ of this matrix describes the contribution of the physical frequency $f_{\text{coeff}}(k)$ (Equation (3.4)) to the spectrum of frame m centered at the physical time instant

$$t_{\text{coeff}}(m) := (mH + B/2) \cdot T = \frac{mH + B/2}{f_s}. \quad (3.9)$$

In Figure 3.5, we show the magnitude spectrograms for the two “Fidelio” examples from Figure 3.7. For computing the STFT, we used a Hamming window with parameters $B = 4096$ and $H = 2048$. As for the waveform, we can roughly estimate the rhythm and loudness from the spectrograms. Furthermore, we observe the decaying behavior of the piano notes in contrast to the sustained horn notes in the Adagio part. The vertical arrangement of the horizontal lines shows some kind of repetition along the frequency axis—caused by the partials (compare Section 2.2). We also see that the higher partials have different amplitudes and individual decay time. Finally, the vertical lines in the piano spectrogram indicate the percussive onsets of the piano hammers. In comparison, the onsets of the orchestra recording seem to be softer.

The STFT cannot reach an arbitrary high resolution in both time and frequency domain at the same time. Related to the Heisenberg uncertainty principle, this is known as the **Fourier uncertainty principle**. To balance out this tradeoff, researchers proposed several **time-frequency transforms**, which are suitable for different purposes. For music processing applications, the **Constant-Q Transform (CQT)** is a useful concept that relates to human auditory perception [27, 217]. In contrast to the STFT, the coefficient's have a logarithmic frequency spacing, which—with appropriate parameters—may correspond to musical pitches.

3.4 Standardized Audio Features

For MIR tasks such as music classification, we need compact representations that capture important characteristics of the audio content while ignoring irrelevant information [154, 237, 264]. Ideally, these **audio features** Θ carry some semantic meaning related to human perception. Sometimes, people categorize the features according to the quality of their semantic meaning. **Low-level** features describe rather technical properties of the signal and often have no direct interpretation. An example is the **Zero Crossing Rate** Θ_{ZCR} , which we obtain by counting the sign changes of the signal in the time domain. **High-level** features have an explicit meaning such as, for example, the key or tempo of a piece. **Mid-level features** relate to human-interpretable concepts but in a way that is not obvious. In this section, we present a selection of standard audio features commonly used for MIR tasks [154, 237, 251]. In the following, we focus on features based on a spectrogram representation. Some of these features originate from the field of speech processing but showed success for processing music data as well. The **Moving Pictures Expert Group (MPEG)** defined a set of such descriptors in the MPEG-7 standard [139]. We roughly follow Peeters [184] who gives an overview of the most important audio features. In Chapter 8, we present classification experiments on the basis of different features types. Since we merely use the standard descriptors for baseline experiments, we only mention the most important concepts and do not focus on technical details.

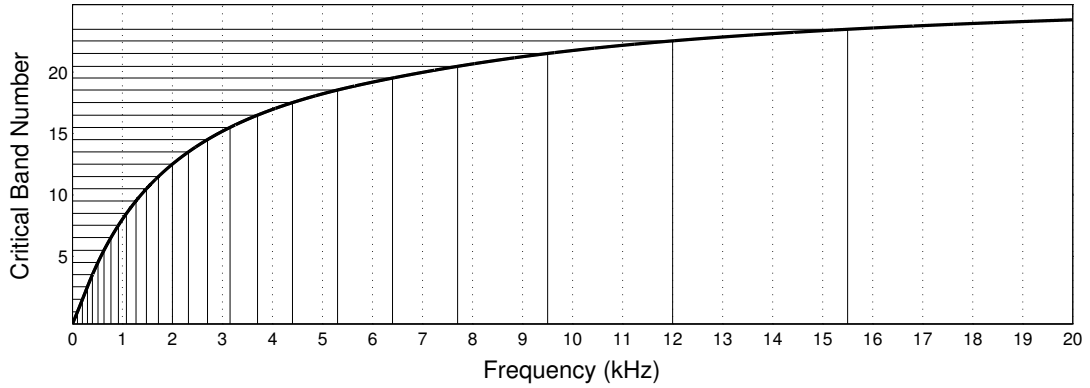
To describe the spectral properties of a signal in more detail, researchers usually compute the features for several **frequency bands** individually. To model human auditory perception, Zwicker [271] proposed a perceptual frequency scale called **Bark scale** (Figure 3.8 (a)). Dividing this scale into equidistant intervals leads to the Bark bands or **critical bands**, which have a particular meaning in the context of psychoacoustics. As a simplification, researchers often use a simple logarithmic scale to derive bands. Figure 3.8 (b) shows such a partitioning where each octave is subdivided into four bands starting at 250 Hz. In the following, we calculate the features for a subset of such frequency bands $j \in [0 : J - 1]$ using different scales. Each band j comprises a set \mathcal{K}_j of STFT frequency coefficients where N_j denotes the total number of coefficients within the band.

A set of features describing the spectral shape is the **Audio Spectral Envelope (ASE)**. From the magnitude spectrogram \mathcal{S} (Equation (3.8)), we obtain the ASE features by summing up the energies within each band j :

$$\Theta_{\text{ASE}}(j, m) := \sum_{k \in \mathcal{K}_j} \mathcal{S}(k, m) \quad (3.10)$$

Here, we use two logarithmic bands per octave from 125 Hz to 16 kHz, together with two bands summarizing the lower and higher frequencies, respectively. In Figure 3.9, we display the audio spectral envelope for the two audio excerpts of the “Fidelio” overture. The broad

a) Critical bands (Bark scale)



b) Logarithmic bands (log scale)

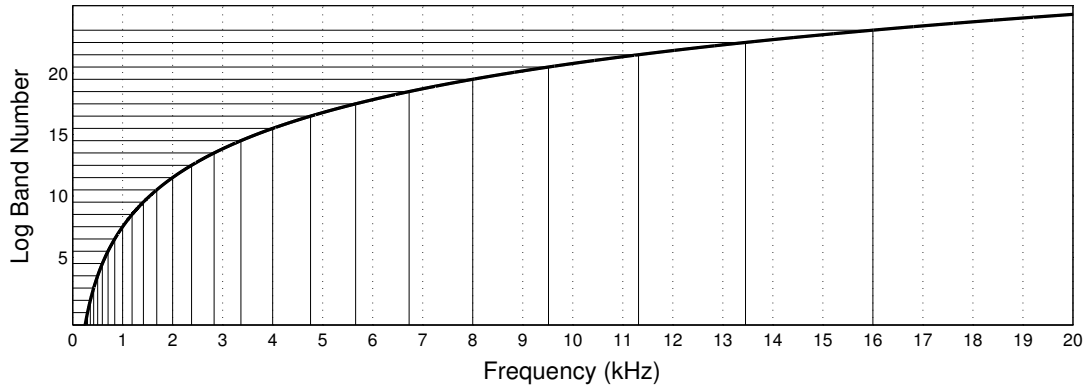


Figure 3.8. Frequency mapping using different scales. In the upper plot, we show the conversion from a linear frequency scale in Hz to the Bark scale indicated by the thick line. Dividing the Bark scale into equidistant intervals, we obtain the critical bands. The lower plot shows an approximation by using a logarithmic scale. We use four bands per octave starting at 250 Hz.

dark area in the beginning phase indicates the wide range of pitches here. In contrast, the second part concentrates on a more specific frequency region. Furthermore, we observe the decays of the piano notes. The broader spectral shape for the horn notes in the orchestra recording—compared to the piano equivalents—may result from having more energy in the higher partials of the horn spectrum.

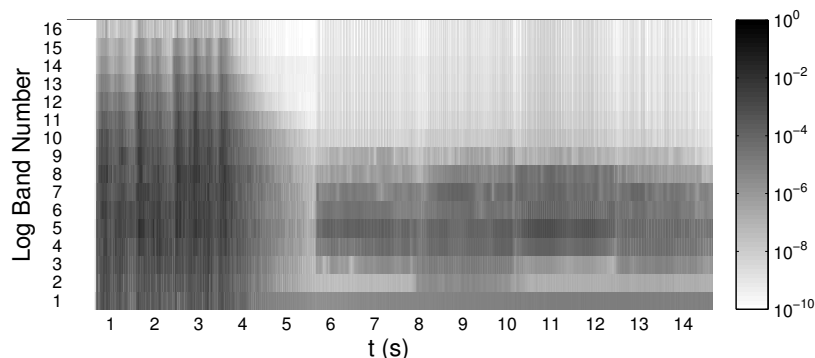
The **Spectral Flatness Measure** (SFM) relates to the noisiness or percussiveness of a signal frame in the respective bands:

$$\Theta_{\text{SFM}}(j, m) := \frac{\left(\prod_{k \in \mathcal{K}_j} \mathcal{S}(k, m)\right)^{1/N_j}}{\sum_{k \in \mathcal{K}_j} \mathcal{S}(k, m)/N_j} \quad (3.11)$$

Small values $\Theta_{\text{SFM}}(j, m)$ occur for tonal frames exhibiting only few sharp frequency components. A related measure is the **Spectral Crest Factor** (SCF) depending on the maximal spectral magnitude

$$\Theta_{\text{SCF}}(j, m) := \frac{\max_{k \in \mathcal{K}_j} \mathcal{S}(k, m)}{\sum_{k \in \mathcal{K}_j} \mathcal{S}(k, m)/N_j}. \quad (3.12)$$

a) Orchestra Recording



b) Piano Recording

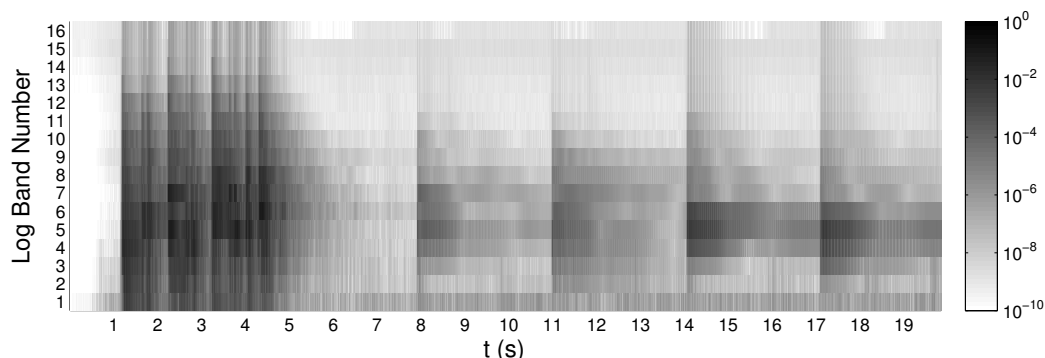


Figure 3.9. Audio spectral envelope features for the “Fidelio” examples. The first band summarizes low frequencies up to 125 Hz, the last band ($j = 16$) comprises high frequencies from 16 kHz to the Nyquist frequency (22 kHz).

The **Spectral Centroid** (SC) estimates the “center of mass” frequency of a frame in each frequency band:

$$\Theta_{\text{SC}}(j, m) := \frac{\sum_{k \in \mathcal{K}_j} \mathcal{S}(k, m) f_{\text{coeff}}(k)}{\sum_{k \in \mathcal{K}_j} \mathcal{S}(k, m) / N_j} \quad (3.13)$$

We calculate Θ_{SFM} , Θ_{SCF} , and Θ_{SC} for 16 logarithmic bands with four bands per octave—comprising a range from 250 Hz to 4 kHz.

A more specialized feature set for describing spectral envelopes are **Mel Frequency Cepstral Coefficients** (MFCC), extensively used for speech processing purposes [26, 158]. To compute these features, we map the frequencies onto the so-called **mel scale**—another perceptual frequency scale derived from human ratings of pitch distances [229]. We group the spectrogram bins into mel bands using triangular filters whose center frequencies are equally spaced over the mel scale (Figure 3.10). From the resulting mel-band amplitudes, we calculate the logarithm and apply the **Discrete Cosine Transform** (DCT). The DCT is a real-valued transform related to the Fourier transform and has several applications in digital signal processing. Performing DCT on the mel-band magnitudes yields some kind of “spectrum of the spectrum”—often denoted with the artificial word “cepstrum.” Usually, researchers take the first 12–16 DCT coefficients as MFCCs. In Figure 3.11, we show an overview of the MFCC calculation procedure. MFCCs turned out useful for several tasks related to musical timbre such as speech-music discrimination [141], music similarity analysis [142], or music genre classification [237].

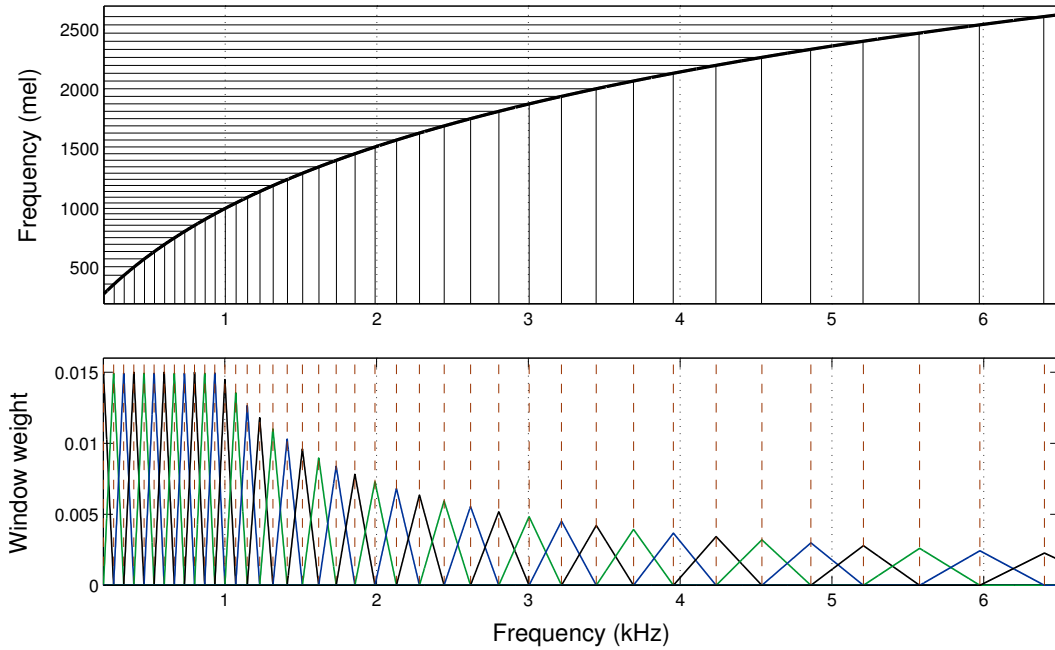


Figure 3.10. Mel scale mapping and triangular filters. The thick line in the upper plot marks the conversion from a linear frequency scale in Hz to the mel scale. The vertical and horizontal lines correspond to the center frequencies of the triangular windows used for calculating MFCCs. In the lower plot, we show these triangular filters. We normalized the windows to a total weight of 1.

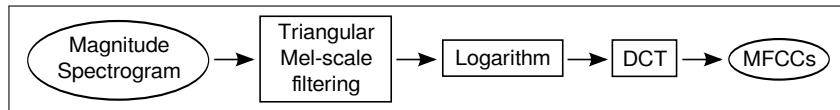


Figure 3.11. Schematic overview of the MFCC calculation.

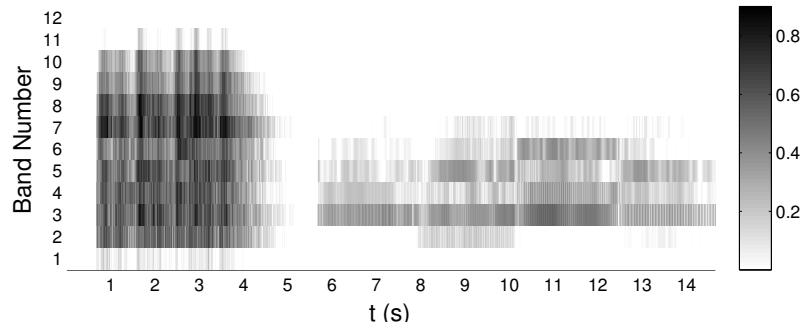


Figure 3.12. Loudness features for the “Fidelio” orchestra excerpt. Here, we plot the feature $\Theta_{\text{LogLoud}}(j, m)$ relating to the specific loudness for 12 critical bands.

For the latter task, Jiang *et al.* [108] proposed a similar but music-specific feature called **Octave Spectral Contrast (OSC)**. To compute these features, they use six logarithmic bands with one band per octave. For each band, they estimate the difference between the spectral peaks and valleys instead of taking the average spectral energy. After taking the logarithm from these differences, Jiang *et al.* apply a Karhunen-Loeve transform instead of a DCT as used for calculating MFCCs.

In addition to these timbre-related descriptors, features for describing the loudness capture useful information. To account for human loudness perception, we weight the frequencies with

the ear transfer curve and group the frequencies into critical bands [60]. From the resulting **specific loudness** for each band, we take the logarithm obtaining the feature $\Theta_{\text{LogLoud}}(j, m)$. In Figure 3.12, we show these loudness values for the “Fidelio” orchestra example. Compared to Figure 3.8 (a), we summarize each two of the critical bands. We can observe the overall loudness shape with the *forte* beginning and the second part in *piano*. In addition to the logarithmized loudness, we obtain a second loudness feature $\Theta_{\text{NormLoud}}(j, m)$ by normalizing the specific loudness for each frame. This results in a relative loudness measure for each band independent from the total loudness.

3.5 Pitch-Based Features

3.5.1 Log-Frequency Spectrogram

The spectrogram \mathcal{S} introduced in Section 3.3 exhibits a linear spacing of the frequency parameters $k \in [0 : K/2]$. In contrast, humans perceive pitch distances in a logarithmic fashion (see Section 2.2). We rate pitch distances as equal that share the same relation of their fundamental frequencies f_0^a and f_0^b . For this reason, we define a logarithmic distance measure

$$\Delta(f_0^a, f_0^b) := \gamma \log \left(\frac{f_0^b}{f_0^a} \right) = \gamma (\log f_0^b - \log f_0^a) \quad (3.14)$$

with a suitable constant γ . For pitches of the twelve-tone equal-tempered scale, we obtain the distance in semitones when setting $\gamma := 12/\log(2)$:

$$\Delta(f_0^a, f_0^b) := 12 \log_2 \left(\frac{f_0^b}{f_0^a} \right) \quad (3.15)$$

This measure Δ is identical to the definition in Equation (2.18) for the corresponding pitches. Because of this perceptual behavior, a logarithmic spacing of the frequencies turned out useful for analyzing harmonic content. Corresponding to the pitch definition in Equations (2.10) and (2.11), we compute the **log-frequency spectrogram** \mathcal{Y} via

$$\mathcal{Y}(p, m) := \sum_{k \in \mathcal{W}_p} \mathcal{S}(k, m). \quad (3.16)$$

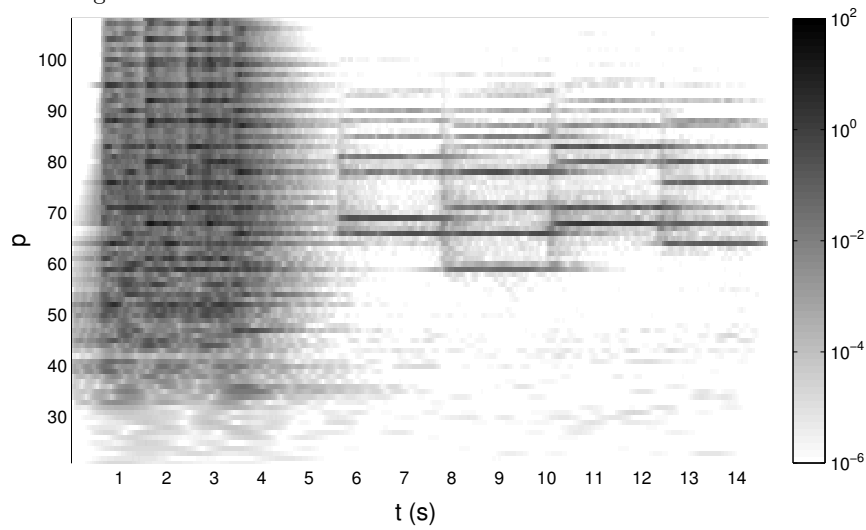
For the pitch p with center frequency $f_0(p)$, we define the set of frequencies

$$\mathcal{W}_p = \{k : f_0(p - d_p) \leq f_{\text{coeff}}(k) < f_0(p + d_p)\} \quad (3.17)$$

with a usual size of $d_p = 0.5$. Here, we use the frequency $f_{\text{coeff}}(k)$ as defined by Equation (3.4). We extend the definition of $f_0(p)$ in Equation (2.11) to continuous values $p \in \mathbb{R}$. By computing \mathcal{Y} (Equation (3.16)), we perform two steps at once. We rescale the frequency axis to a **logarithmic spacing** and sum all neighboring frequencies that belong to a pitch p (**frequency binning**). Therefore, the rows of \mathcal{Y} correspond to the musical pitches on an equal-tempered scale.¹¹ Because of that, \mathcal{Y} is also denoted as **pitchogram**. Inverting Equation (2.11), we see the logarithmic frequency spacing for a linear series of pitch numbers:

¹¹Note that this only relates to the spacing of the frequency axis. The log-frequency spectrogram does not reflect the perceptual phenomenon of pitch as a compound sound of a series of partials.

a) Orchestra Recording



b) Piano Recording

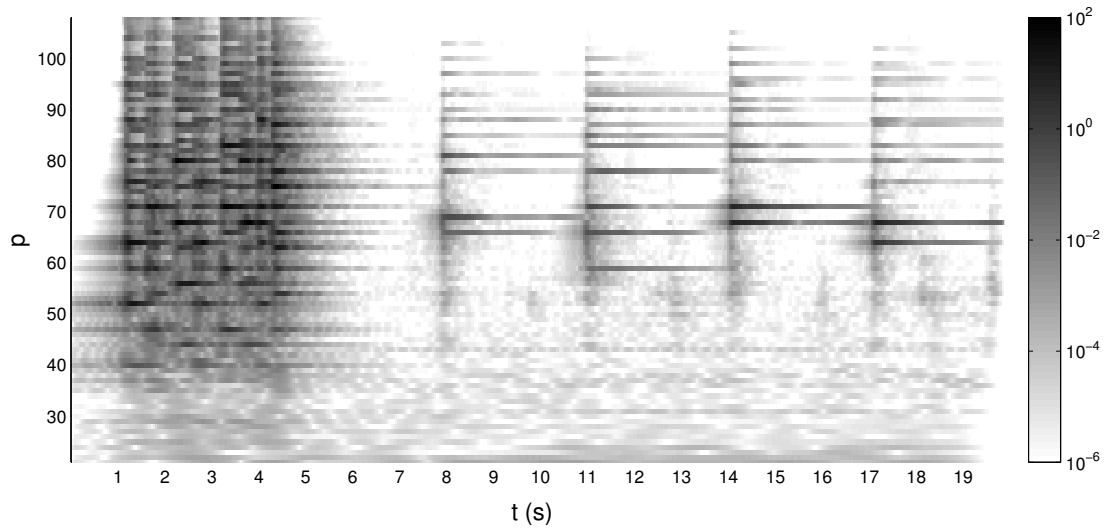


Figure 3.13. Log-frequency spectrograms of the two “Fidelio” examples. We computed these spectrograms using a bank of elliptic filters as published in [165]. The frame specifications (blocksize B and hopsize H) are identical to the linear-frequency spectrograms shown in Figure 3.7. We display the pitches of the piano range $p \in [21 : 108]$. The gray levels indicate the energy values in the pitch bands.

$$p(f) = 12 \log_2 \left(\frac{f}{f_{\text{concert}}} \right) + 69. \quad (3.18)$$

for $f \in \mathbb{R}^+$.

The procedure presented above constitutes a simple filtering with center frequencies $f_0(p)$ and a bandwidth $\Delta_{\text{BW}} \in \mathbb{R}$ of

$$\Delta_{\text{BW}}(p) := f_0(p + d_p) - f_0(p - d_p). \quad (3.19)$$

With Equation (2.11), we obtain

$$\Delta_{\text{BW}}(p) = \left(2^{d_p} - 2^{-d_p}\right) \cdot f_0(p) \quad (3.20)$$

Therefore, the bandwidth decreases towards lower pitches. Together with a linear spacing of the frequency parameters $k \in [0:K/2]$, this may lead to a poor resolution for the lower pitches since the set \mathcal{W}_p may comprise only few or even zero frequency coefficients k (Equations (3.4) and (3.17)). This effect is exceptional for a low frequency resolution of the STFT.¹² For this reason, scholars proposed several approaches for improving the frequency resolution for the whole range of musically relevant pitches. A popular method is the constant-Q transform (Section 3.3), which turned out useful for several audio analysis purposes [194, 268]. It is convenient to directly space the constant-Q filters in semitones.

Another method to improve spectral resolution relies on a reassignment of the time and frequency coordinates. This approach incorporates phase information by using the phase derivative from the complex-valued spectrogram. By reallocating the spectral energy, we obtain an **Instantaneous Frequency** (IF) spectrum [1, 2]. Several feature implementations for describing musical pitch rely on this time-frequency transform [57, 115]. As a further strategy, Müller *et al.* [161, 165, 170] use a **multi-rate filter bank** of elliptic filters to account for the different pitch ranges.

For Figure 3.13, we used the latter approach to compute the log-frequency spectrogram for the two “Fidelio” examples. In the second part of the examples (Andante), we now can observe the interval structure of the horn motif. Due to the logarithmic spacing, the frequencies of the overtones have less distance in the higher regions. Comparing the two examples, we again see the more percussive attacks as well as the decaying character of the notes in the piano recording. Besides the partials, many more frequencies contribute with small but non-zero energies. Due to the different sampling rates for the filters, the lower pitches show a coarser time resolution. This constitutes the tradeoff of a sufficiently high frequency resolution in the low range.

3.5.2 Chroma Features

In Section 2.2, we outlined the special role of octave relationship for human pitch perception. For analyzing harmonic phenomena, representing the pitch class content of the music came out beneficial. Researchers proposed methods for extracting pitch class information from audio using signal processing methods [17, 18, 67, 167]. Usually, these features are called **pitch class profiles** or **chroma features**. A chroma vector $\mathbf{c} := (c_0, c_1, \dots, c_{11})^T \in \mathbb{R}^Q$ of dimension $Q := 12$ describes the energy of the pitch classes $q \in [0 : Q - 1]$. We adopt the definition in Equation (2.13) with $q = 0$ denoting the pitch class C, and so on:

$$(0, 1, \dots, 11) \hat{=} (C, C\sharp, \dots, B) \quad (3.21)$$

From the log-frequency spectrogram \mathcal{Y} , we obtain one chroma entry c_q by summing up the energy of all pitches $\{p \mid p \bmod 12 = q\}$ belonging to this pitch class q . The series of chroma

¹²Typically, K is equal to the blocksize B (number of samples per STFT frame). In this case, there is a tradeoff between time and frequency resolution of the log-frequency spectrogram.

vectors for the frames $m \in [0 : M - 1]$ forms a **chromagram** \mathcal{C} defined by

$$\mathcal{C}(q, m) := \sum_{\{p | p \bmod 12 = q\}} \mathcal{Y}(p, m). \quad (3.22)$$

One column of the chromagram corresponds to the chroma vector $\mathbf{c}^m := \mathcal{C}(\cdot, m)$ for a fixed frame index m . For analyzing the harmonic content of an audio recording, we are only interested in the *relative* energy of the values. To this end, we normalize the chroma vectors. Mathematical norms typically used for this purposes are the **Manhattan norm**

$$\ell_1(\mathbf{c}) := \sum_{q=0}^{Q-1} |c_q| \quad (3.23)$$

or the **Euclidean norm**

$$\ell_2(\mathbf{c}) := \left(\sum_{q=0}^{Q-1} c_q^2 \right)^{1/2}. \quad (3.24)$$

Using one of these norms $\ell_z(\mathbf{c})$, we replace every chroma vector \mathbf{c} with its normalized version

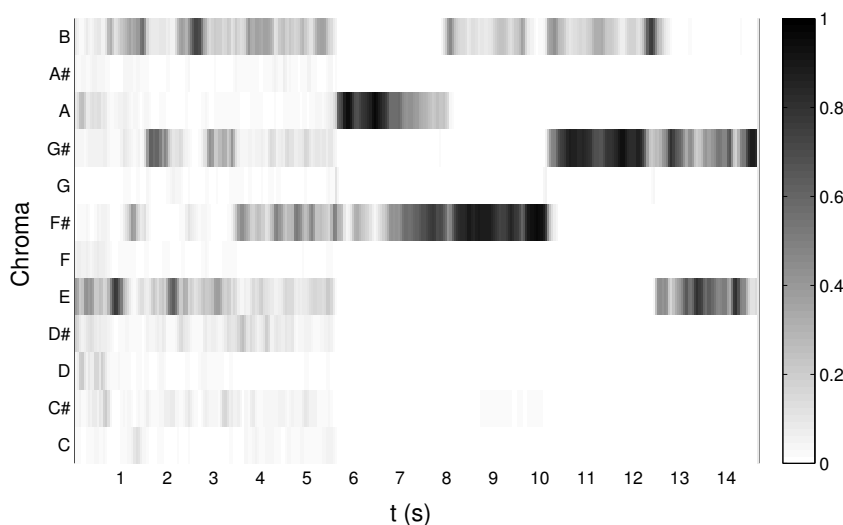
$$\mathbf{c}^{\ell_z} = (c_0^{\ell_z}, \dots, c_{11}^{\ell_z})^T := \frac{\mathbf{c}}{\ell_z(\mathbf{c})}, \quad (3.25)$$

obtaining the **normalized chromagram** \mathcal{C}^{ℓ_z} . For frames with very low energy, the normalization process may lead to random-like chroma vectors. To avoid artifacts in the normalization step, some authors introduce an **energy threshold** ε and assign a flat vector to the respective frames [165, 167]. Conceptually, applying column-wise normalization corresponds to some kind of dynamic equalization by ignoring characteristics such as overall energy or loudness. For analyzing harmonic effects such as the occurrence of certain chords, we are only interested in the *relative* pitch class importance independently of the signal's local energy.

In Figure 3.14, we show the normalized chromagrams \mathcal{C}^{ℓ_1} for the orchestra and piano recording of the “Fidelio” opening. For this, we used the public **Chroma Pitch** (CP) implementation based on elliptic filters [165]. We clearly observe the pitch classes from the unisono melody (first half) and the horn motif (second half). In such a chroma representation, we cannot resolve the difference between the notes B3 and B4 in the horn motif since these pitches belong to the same pitch class. Comparing the two versions, we find a very similar structure, in general. Differences occur with respect to the balance within chords or intervals. Looking at the P5 interval B–F \sharp (at about 9 s in the orchestra recording and 12 s in the piano version), we find a more equal energy balance in the piano version.

Because of the normalization, this chromagram does not capture the decay phases of the piano chromagram as observed in the other representations (Figures 3.7 and 3.13). The light gray area in the piano chromagram (at about 7 s) has energy values below the threshold ε and, thus, obtains a flat chroma distribution. Overall, we can see that chroma features are much more robust against variations in timbre or loudness compared to spectrogram representations. However, the prominent pitch classes in the chromagrams do not exactly correspond to the notes in the score. One reason is the presence of overtones. This leads to some energy contribution for pitch classes corresponding to the overtones of the played notes rather than to their fundamental. The pitch class F \sharp at about 1.5 s in the orchestra chromagram of Figure 3.14 may be an example for such an effect. In the corresponding measure in

a) Orchestra Recording



b) Piano Recording

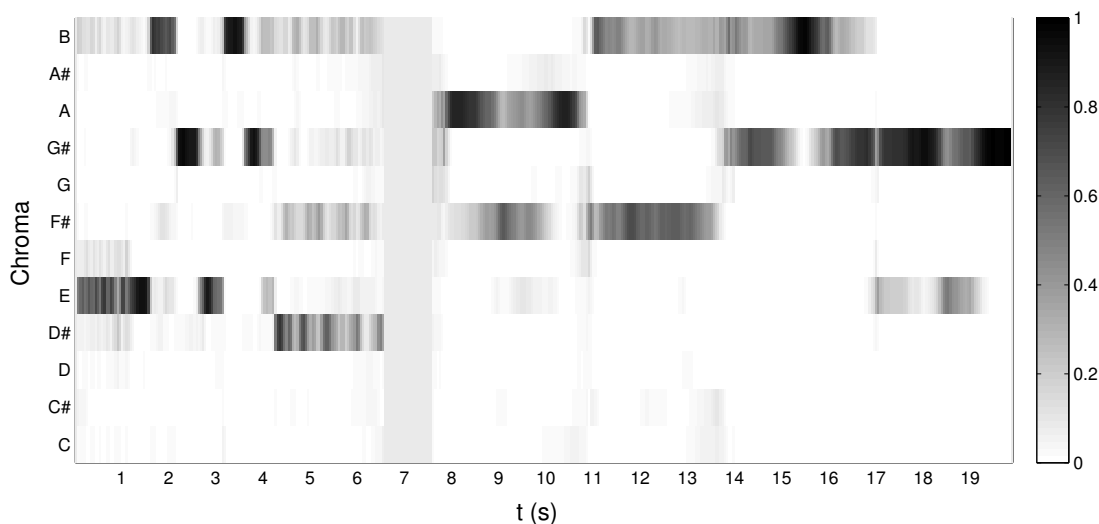


Figure 3.14. Chromagrams of the two “Fidelio” recordings. Here, we show the normalized chromagram version C^{ℓ_1} computed with a public implementation based on elliptic filters [165]. The frame specifications correspond to the Figures 3.7 and 3.13. On the vertical axis, we specify the note names corresponding to the chromagram’s rows. We encode the chroma values via different gray levels.

the score, there is no $F\sharp$ note. However, the third partial of the prominent note B corresponds to this pitch class. Apart from such problems, percussive effects or percussion instruments may deteriorate the chroma features’ clarity due to their broad frequency distribution. This constitutes a major problem for analyzing harmonies from popular music, which typically includes drums as a standard instrument.

Table 3.1. Different methods for extracting chroma features from audio. Apart from the different time-frequency transforms, the features vary with respect to different pre- and post-processing techniques.

<i>Authors</i>	<i>Name</i>	<i>Transform</i>	<i>Specifications</i>	<i>Application</i>
Fujishima [67]	PCP	STFT	–	Chord recognition
Bartsch & Wakefield [17]	–	STFT	Beat synchronization	Audio thumbnailing
Bello & Pickens [21]	–	CQT	Beat synchronization	Chord recognition
Gómez [76–78]	HPCP	STFT	Spectral peak-picking, overtone estimation	Global key detection
Lee [131]	EPCP	CQT	Overtone estimation	Chord recognition
Ellis & Poliner [57]	IFC	IF	Beat synchronization	Cover song identific.
Ueda <i>et al.</i> [238]	FTC	CQT	Harmonic-percussive separation, diagonalization	Chord recognition
Müller & Ewert [164, 166]	CRP	Elliptic filters	Log. compression, timbre homogenization (DCT)	Audio matching
Müller <i>et al.</i> [167]	CENS	Elliptic filters	Logarithmic quantization, temporal smoothing	Audio matching
Mauch & Dixon [147]	NNLS	STFT	Spectral windowing, approximate transcription	Chord recognition
Khadkevich <i>et al.</i> [115]	RC	IF	–	Chord recognition
Kronvall <i>et al.</i> [123]	CEBS	–	Sinusoidal modeling, sparsity constraints	Visualization

3.5.3 Timbre Invariance and Enhanced Chroma Features

3.5.3.1 Overview

Researchers proposed several approaches to overcome problems as described above and to boost the invariance of chroma features against timbral variations. In general, the benefit of certain chroma improvement strategies considerably depends on the specific application context. A number of authors focused on **chord labeling** as application and compared different chroma features for this purpose [36, 109, 147, 228]. Another case for applying chroma-based strategies is **audio matching**. In [164], the authors evaluated several chroma feature implementations with respect to this application. In the following, we will present the most important ideas to improve the robustness of chroma features against timbral variation and signal processing artifacts. Table 3.1 gives an overview of these contributions together with the applications used for testing the respective features.

3.5.3.2 Overtone Removal

As we mentioned previously, an important deficiency of simple chroma extraction methods is the influence of overtones belonging to pitch classes other than the fundamental’s pitch class. To reduce these contributions, Gómez proposed a strategy for estimating the overtones using a geometric decay model for the amplitudes $a(h) \in \mathbb{R}$ [76]

$$a(h) := s^h \tag{3.26}$$

for $h \in \mathbb{N}$ being the partial number and $s \in]0, 1[$. For the latter parameter, Gómez proposed a value of $s = 0.6$. Considering the harmonic partials this way, we obtain the **Harmonic**

Pitch Class Profiles (HPCP). To approach the same problem, Lee [131] proposed a method using the **Harmonic Product Spectrum (HPS)**. For computing the HPS, we multiply each frequency coefficient in the spectrogram with several components corresponding to integer multiples of this frequency. This leads to a reduction of non-tonal elements in the features resulting in the **Enhanced Pitch Class Profiles (EPCP)**.¹³ A further method by Mauch *et al.* [147] makes use of idealized note profiles. These profiles follow a geometric decay as described by Equation (3.26), with a suggested value of $s = 0.7$ for popular music. The authors obtain a fundamental frequency pitchogram by solving a **Non-Negative Least Squares (NNLS)** problem, which minimizes the squared differences between the log-frequency spectrogram \mathcal{Y} and the aggregated note profiles. This pitchogram extraction constitutes some kind of **approximate transcription** and builds the basis for the NNLS chroma feature computation.

3.5.3.3 Timbre Homogenization

Apart from these overtone removal strategies, several researchers proposed ideas to homogenize the timbre by flattening the spectral envelope. A common procedure to do this is **spectral whitening**, which removes short-time correlation from the signal by locally normalizing the subbands [117,118]. The HPCP feature computation incorporates such a step [76]. Müller and Ewert proposed another strategy for flattening the spectral envelope [164,166]. Their method relates to the computation of MFCCs (Section 3.4) but uses a pitch scale instead of the mel scale before applying the DCT. From the resulting **Pitch Frequency Cepstral Coefficients (PFCCs)**, they discard the lower ones that relate to timbral characteristics as described by the spectral envelope. After performing the inverse DCT, the resulting pitch bands are mapped onto chroma values. The resulting features are called **Chroma DCT-Reduced Log Pitch (CRP)**. Because of the PFCC elimination, negative CRP values can occur after applying the inverse DCT.

As a simpler strategy to reduce the influence of timbral characteristics, some authors perform logarithmic compression before the chroma mapping step [119,166,238]. For this purpose, we replace the log-frequency spectrogram \mathcal{Y} describing the energy per pitch band with a logarithmized version

$$\mathcal{Y}_{\log}(p, m) := \log(1 + \eta \cdot \mathcal{Y}(p, m)) \quad (3.27)$$

with a parameter $\eta \in \mathbb{R}^+$. Typical values from the literature are $\eta = 100$ or $\eta = 1000$ [36,164,166]. Computing chroma features on the basis of \mathcal{Y}_{\log} , we obtain the **Chroma Log Pitch (CLP)**.

3.5.3.4 Other Enhancement Strategies

Overtone and timbral properties mostly contribute to the high pitch regions. Furthermore, the very low pitches suffer from a bad frequency resolution in the time-frequency transform, in many chroma implementations.¹⁴ Due to these effects, a simple reduction of the pitch range for the chroma computation may already improve the feature quality. A typical selection is the pitch range of the piano [164]. To weaken the effect of the outer frequency regions,

¹³The authors suggest to take only frequency multiples of powers of two. In this case, only octave-related partials with the same pitch class contribute to the HPS.

¹⁴For popular music including drums, an additional effect arises from the bass drum, which often contributes with a particular pitch to the spectrogram representations [36].

some authors introduce a Gaussian window for weighting the pitches of the log-frequency spectrogram [36, 147, 163] centered, for example, at the note C4 with $p = 60$

$$\mathcal{Y}_W(p, m) := \exp\left(-\frac{(p-60)^2}{2 \cdot 15^2}\right) \cdot \mathcal{Y}(p, m). \quad (3.28)$$

Some authors also use a second window covering the lower octaves only (centered at about $p = 40$) to obtain a **bass chromagram** [58, 147, 148, 163, 208]. Combining bass and treble chromagrams, an estimation of chord inversions is possible.

For computing the HPCP features, Gómez proposes further enhancement strategies. To reduce spectral noise, she applies a spectral peak-picking stage prior to the overtone estimation [76]. Another problem with chroma features arises from non-tonal frames such as transients or percussive events. In the HPCP extraction procedure, a **transient location** method removes these frames prior to the time-frequency transform [23, 76]. To account for these percussive components, other researchers experimented with **Harmonic-Percussive Source Separation** algorithms such as [176] as a preprocessing stage [173, 238].

3.5.3.5 Comparison of Chroma Types

Implementations of several chroma extraction methods are publicly available. The Chroma Toolbox¹⁵ comprises MATLAB implementations of the feature types CP, CLP, and CRP [165]. For extracting HPCP features¹⁶ and NNLS features,¹⁷ Vamp plugins for the use with open source software such as Sonic Visualizer¹⁸ or Sonic Annotator¹⁹ are accessible online. For the EPCP features, we use a re-implementation of the method described in [131].

In Figure 3.15, we show the chromagram of the “Fidelio” orchestra examples for different chroma extraction methods. For the CLP features, we clearly observe the enhancement of the weaker components through logarithmic compression, especially for the *forte* beginning. Here, timbre homogenization with methods such as CRP helps to remove non-harmonic noise. The EPCP features show a contrary behavior since they suppress weak components. This leads to a sharper description of the fundamentals. On the other hand, we see more fluctuations in the chroma structure. Furthermore, pitches with less energy in the respective overtones almost disappear. We observe such problems for some of the horn notes in the second half of the example already with only two HPS iterations. The NNLS method seems to conduct a more careful overtone removal. Here, we find no suppression of played pitches but overtones such as the D \sharp between 10s and 12s (third harmonic of G \sharp) obtain smaller values. Additionally, the least squares overtone estimation also leads to some enhancement of weak components (as similarly described in [36]).

In the respective publications, the authors tested their proposed chroma extraction methods with respect to a particular application (see Table 3.1). Furthermore, there are several studies dedicated to a comparison of chroma feature performance. Stein *et al.* [228] conducted a comparison experiment between the feature types PCP (with different weighting functions around a pitch’s center frequency), HPCP, EPCP, IFC, a constant-Q based approach, and a filter bank approach. They measured the difference between the played pitches of synthesized

¹⁵<http://resources.mpi-inf.mpg.de/MIR/chromatoolbox>

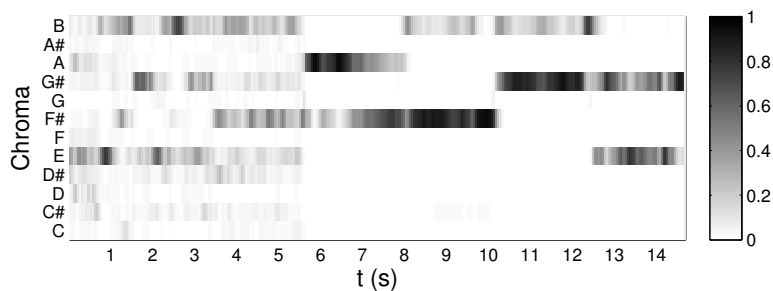
¹⁶<http://mtg.upf.edu/technologies/hpcp>

¹⁷<http://isophonics.net/nnls-chroma>

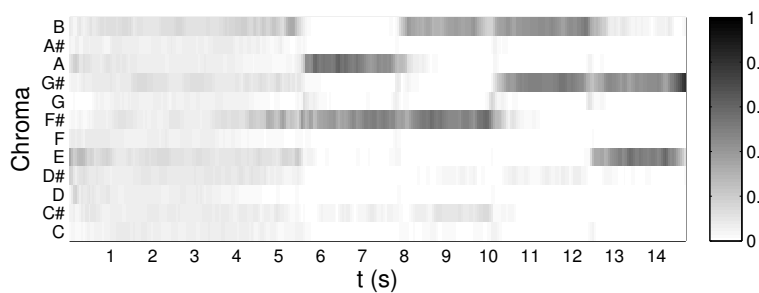
¹⁸<http://www.sonicvisualiser.org> [30]

¹⁹<http://www.vamp-plugins.org/sonic-annotator>

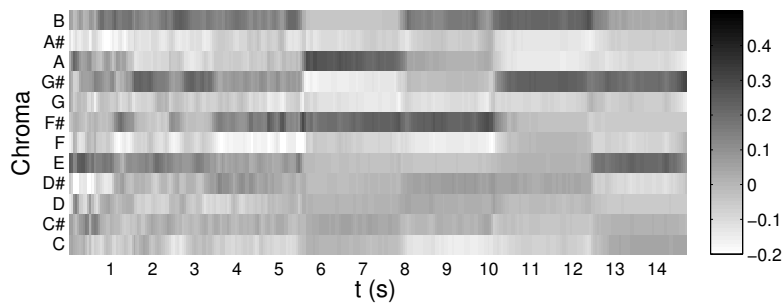
a) CP features



b) CLP features

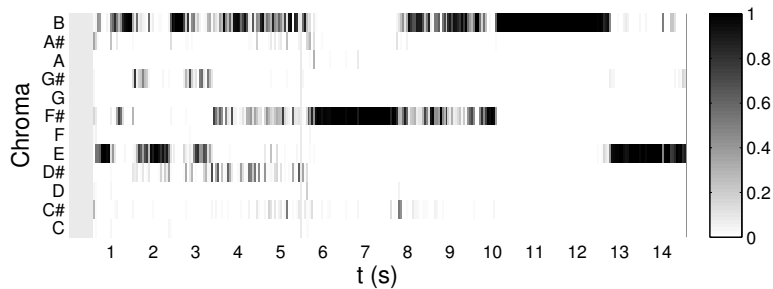
with $\eta = 1000$ 

c) CRP features

coefficients < 55 set to zero

d) EPCP features

2 HPS iterations



e) NNLS features

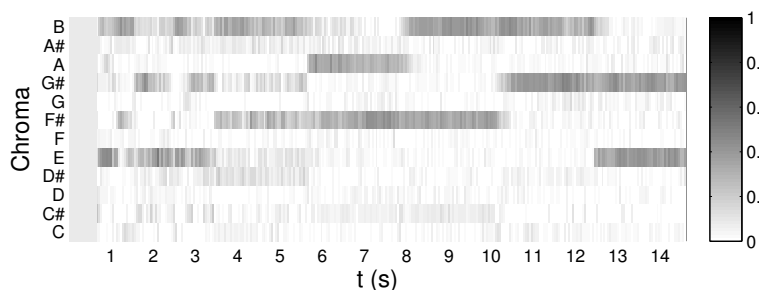


Figure 3.15. Different chromagram representations of the “Fidelio” orchestra recording, first measures. We normalized all chromagrams to C^{ℓ_1} for a direct comparison.

audio excerpts with the chroma values corresponding to those pitches. In this study, EPCP features performed best.

In [123], the authors conducted a similar evaluation by visually comparing the output of two CLP feature types with their own implementation using sparsity constraints. With respect to such evaluations, chroma strategies that suppress weaker components always achieve best results. However, emphasizing weak components such as timbre homogenization turned out useful when used in a real application context. In [164], CRP features led to preferable performance in an **audio matching** experiment based on dynamic time warping. Here, the comparison between features of the same type is important rather than their correspondence to the notated pitches.

Several studies focus on the impact of chroma feature quality for chord recognition. Jiang *et al.* [109] evaluated different filter bank chroma features such as CP, CLP, and CRP as well as an IFC implementation in a chord recognition experiment. They used a chord recognition algorithm based on Hidden Markov Models (HMMs) and evaluated on the Beatles songs with publicly available chord annotations. In this context, logarithmic compression—which is part of both CLP and CRP feature strategies—lead to strong increase in chord recognition performance. On the same dataset, Mauch and Dixon [147] compared their NNLS chroma features against a standard method for chord recognition. They found considerable improvements with NNLS chroma, especially for the detection of difficult chords such as seventh chords or triad inversions.

Cho and Bello [36] published a large study of chord recognition algorithms evaluated on a dataset of about 500 pop songs. They re-implemented several chroma extraction algorithms presented here such as the NNLS and the CRP methods. In this experiment, overtone removal turned out beneficial for chord detection performance. In contrast, the effect of timbre homogenization was small or negative. However, features with a combination of both ideas achieved the best results. For both steps, the simpler approaches performed similar or even better than their complex equivalents. Therefore, overtone removal with a Gaussian filter over the pitch range (Equation (3.28)) seems to be sufficient as well as timbre homogenization with logarithmic compression only. Harmonic-percussive source separation did not lead to improvements in this study.

When comparing chord detection experiments, the selection of chord types considered for detection (and evaluation) is an important factor. Using NNLS chroma, Mauch *et al.* [147] observed considerable improvements for difficult chords. In contrast, others only consider major and minor triads together with a *No chord* state, which is a simplified scenario that cannot properly describe all harmonic phenomena in pop songs [36, 109].

3.5.4 Tuning Estimation

As we discussed in Section 2.4, instruments or ensembles may employ a global tuning other than the standard concert pitch $f_{\text{concert}} = 440$ Hz. In particular, historical performances of Early Music make use of lower global tuning. When using fixed center frequencies for the filter banks or the pitch summarization, this may lead to problems in the chroma computation. In the worst scenario, a played pitch contributes with equal energy to a chroma value and its neighbor, thus leading to a smearing across chroma bands. To avoid such problems, several researchers implement a global tuning estimation into their chroma extraction algorithms. Harte and Sandler [87] propose a finer chroma resolution of 36 bins per octave corresponding to three bins per semitone. Gómez [76] and Lee [131] follow this idea. To adapt to the recording, they consider the twelve bins maximizing the overall energy. Zhu and Kankanhalli

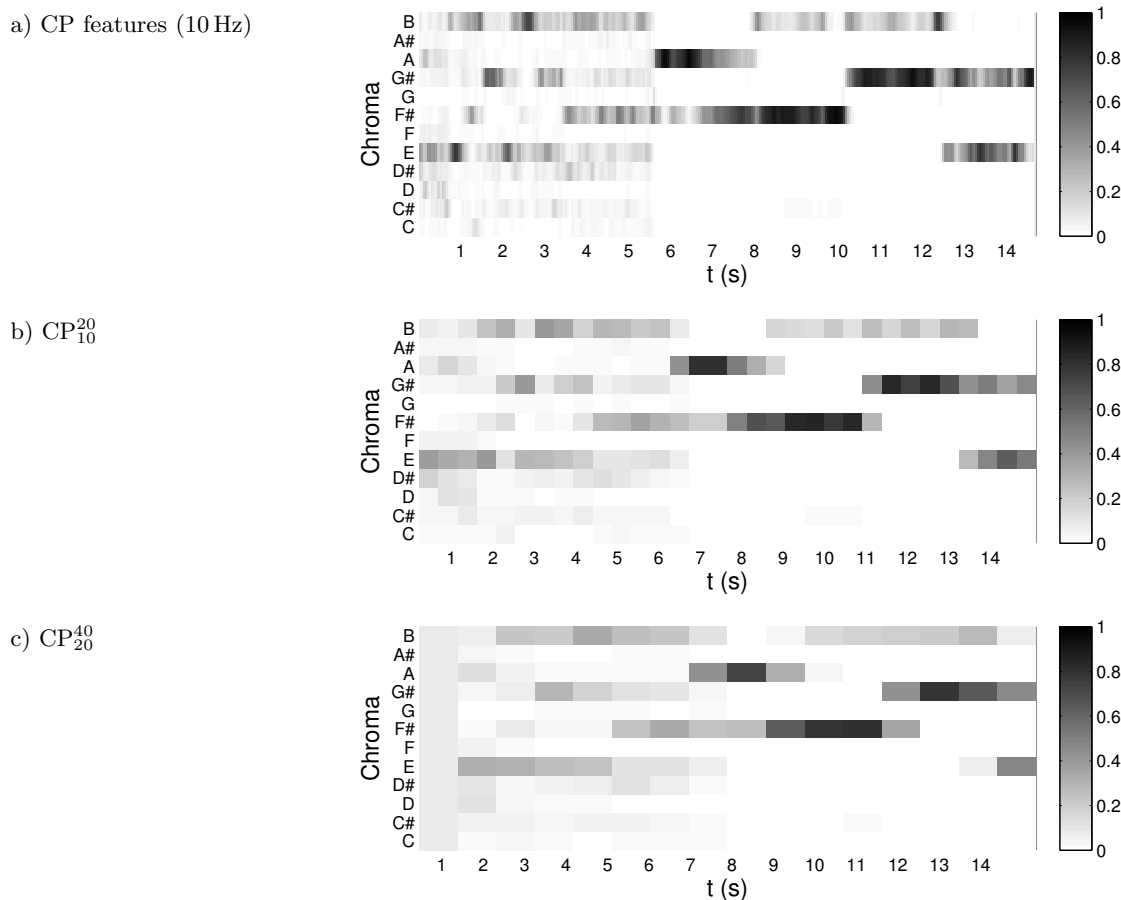


Figure 3.16. Chromagram in different temporal resolutions for the “Fidelio” orchestra recording, first measures. We compute the smoothed versions from a CP chromagram with an initial feature rate of $f_{\text{feat}} = 10$ Hz. Finally, we normalize all chromagrams to C^{ℓ_1} for a direct comparison.

[268] follow a similar idea and choose the energy maximizing band out of ten bands per semitone (± 50 Cent). Müller and Ewert [165] use a similar estimation strategy. Depending on the estimated reference frequency, they use the best out of six shifted filter banks. We follow this approach but use a shifted filter bank only for deviations > 15 Cent from a 440 Hz tuning. Because of performance practice for classical music, we assume all deviating reference frequencies to lie below 440 Hz.

3.5.5 Temporal Resolution and Feature Smoothing

In addition to the chroma enhancement strategies presented in the previous section, we can increase the robustness of chroma representations by locally smoothing the features in a post-processing step. This makes the features invariant against local variations such as articulation or ornamentation. At the same time, the features obtain some different meaning since they describe pitch class statistics rather than the local pitch classes. For a rather fine resolution, this statistics may correspond to local tonal items such as chords. On a coarser scale, concepts such as local keys and modulations may have considerable influence on the smoothed features.

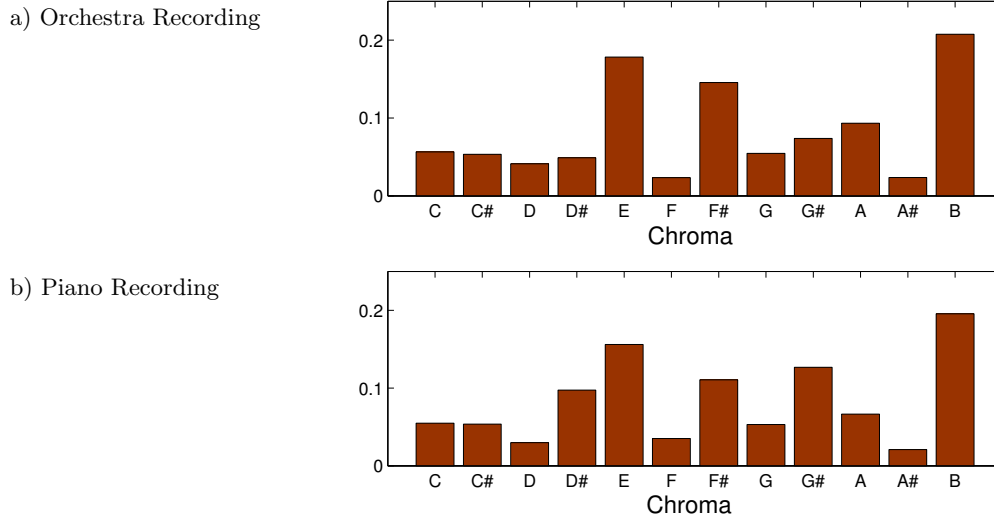


Figure 3.17. Chroma histograms of the two “Fidelio” recordings. We show the normalized histograms \mathbf{g}^{ℓ_1} for both the orchestra and the piano version. In contrast to the previous visualizations, we used the full pieces for computing these histograms instead of only the first measures.

We follow the procedure for computing **Chroma Energy Normalized Statistics** (CENS) as described in [161,167]. We use the implementation of the Chroma Toolbox [165] but leave out the quantization step, which may lead to a loss of information in our application scenarios. For the smoothing process, we consider a selection of $w \in \mathbb{N}$ frames from the original chromagram \mathcal{C} weighted with a Hanning window. We obtain the smoothed chromagram $\mathcal{C}^w(q, m)$. Since the window of length w usually comprises several consecutive chroma vectors, neighboring frames in \mathcal{C}^w exhibit a high degree of similarity. Therefore, we can downsample this sequence by a factor $d \in \mathbb{R}$ by keeping only every d -th vector (with $1 < d < w$). We finally obtain a chromagram $\mathcal{C}_d^w(q, m)$, $m \in [0 : (M_{\text{red}} - 1)]$ of reduced size $M_{\text{red}} \approx \lfloor M/d \rfloor$. In Figure 3.16, we show the initial CP chromagram of the “Fidelio” orchestra example together with two smoothed and downsampled versions \mathcal{C}_d^w for different parameters w and d .

Beyond such smoothed chromagrams, we obtain a very rough summary of a piece’s tonal content by computing a **global chroma histogram** $\mathbf{g} := (g_0, \dots, g_{11})^T \in \mathbb{R}^{12}$ over the whole recording:

$$g_q := \sum_{m=0}^{M-1} \mathcal{C}^{\ell_1}(q, m) \quad (3.29)$$

By analogy with the definition in Equation (3.25), we obtain a normalized histogram \mathbf{g}^{ℓ_1} :

$$g_q^{\ell_1} := \frac{g_q}{\ell_1(\mathbf{g})} \quad (3.30)$$

In Figure 3.17, we show the normalized histograms \mathbf{g}^{ℓ_1} for the full recordings of the “Fidelio” overture. The highest bars correspond to the most important pitch classes in the E major key. Interestingly, the pitch class B (dominant note) shows higher intensity than the tonic note E in both recordings. This may have several reasons. First, B is part of both chords EM (tonic chord) and BM (dominant chord). Furthermore, modulations to the upper fifth key (here B major) are the most frequent ones for pieces in sonata form. Last, B is also part of the overtone series of E, which may lead to further enhancement. Due to this large variety

of effects, a global chroma histogram does not provide enough information for resolving all tonal properties of a music recording. On the other side, such histograms constitute robust and compact representations of musical pieces. Comparing the orchestra histogram with the piano histogram, we see a very similar structure. We only find subtler differences such as the energy of the pitch class $G\sharp$ being more pronounced for the piano recording. This difference may arise mostly due to acoustic behavior such as timbral characteristics of the instruments or the specific instrumentation of the two versions.

3.5.6 Properties of Chroma-Based Analysis

In the previous sections, we showed the efficiency of chromagrams for describing the tonal content of audio recordings. However, the benefits of chroma features come along with a considerable loss of information. In this section, we discuss several important points that we have to consider when using chroma features for tonal analysis.

A fundamental problem of audio-based analyses is the separation of the audio signal into **musical voices**. For ensembles with different instruments, automatic source separation techniques can be useful to approach this problem. For polyphonic music in a monotimbral instrumentation—such as a fugue for piano, organ, or string orchestra—separation is often not feasible since all voices have similar timbral characteristics. It is also hard to separate voices on the basis of fixed pitch ranges since they may exhibit larger jumps and intersect with each other. Hence, we can only estimate to which voice a note event belongs, for example, by considering knowledge about melodies such as the fugue subject. Without a reliable separation of the pitch content into musical voices, it is not possible to automatically analyze voice leading phenomena, which constitute important style characteristics according to [129].

An important step for the chroma computation is the summarization of neighboring frequencies, which belong to a certain pitch (see Equation (3.17)). Thereby, we smooth out subtler differences in pitch and lose the possibility of resolving **details of intonation** and local tuning as well as the information of **enharmonic spelling**. These details may carry some stylistic information since musicians adapt their intonation behavior to the musical style, especially for recordings in historical performance practice. Concretely spoken, this is the computation step where we map all pitches onto the equal-tempered scale. Therefore, we cannot discriminate between enharmonically equivalent pitches such as $G\sharp$ and $A\flat$ on the basis of pitch or chroma features. This observation extends to other harmonic concepts such as intervals. For example, we cannot resolve any difference between a +2 and a m3 interval in such a pitch representation.

Furthermore, we lose information by summarizing octave-related pitches to obtain chroma features. Since we only keep the pitch class instead of the complete pitch information, we have no indications about interval or triad **inversions**. As an example, a chroma vector with strong C and E values may refer to an M3 interval. In the same way, this chroma vector can describe the complementary interval m6 depending on which pitch class belongs to the higher pitch. Therefore, we can only discriminate six different interval categories as shown in Table 2.3. In a melodic context, this limitation also relates to the **direction** of intervals.

For the reasons stated above, it is not possible to apply all concepts of musicological analysis—as presented in Chapter 2—for analyzing audio recordings. We could approach some of the limitations described above by means of more complex algorithms such as source separation and automatic transcription methods. However, most of these algorithms considerably depend on characteristics of the analyzed instruments such as onsets or timbre.

Because of that, these algorithms show deviating results when analyzing recordings with different orchestration. With respect to such properties, analyses based on normalized chroma representations show a higher degree of stability across different interpretations and instrumentations.

3.6 Machine Learning Methods

3.6.1 Experimental Design

In the last decades, automatic methods from the **machine learning** field showed success for analyzing and organizing large databases [5]. In this section, we summarize some relevant techniques that we later apply to audio datasets of Western classical music (Chapters 7 and 8). The main contribution of this thesis lies in the design of new tonal features for classification. For this reason, we are not interested in the technical details of the classification algorithms and rather use them as some kind of “black boxes.” This is why we keep the explanation very brief and confine ourselves to mention only those parameters that are relevant for our experiments.

In general, there are two types of machine learning algorithms. **Unsupervised** learning strategies serve to find structure in unlabeled data. In contrast, **supervised** algorithms learn a mapping from training data to corresponding output values. For discrete output variables, we speak of a **classification** task. As opposed to this, a **regression** problem exhibits continuous output values. Since we do not use regression methods in this thesis, we refer to [5] for interested readers.

As the input to these methods, we have a set of $I \in \mathbb{N}$ examples—the **instances**. For each instance with index $i \in [1 : I]$, we compute a **feature vector** $\Phi^i := (\phi_1^i, \dots, \phi_D^i)^T \in \mathbb{R}^D$ of dimensionality $D \in \mathbb{N}$, which quantifies the characteristics of this instance. Often, the corresponding space \mathbb{R}^D is called **feature space**. The set of feature vectors for all instances forms the **feature matrix** $\mathcal{F} \in \mathbb{R}^{D \times I}$:

$$\mathcal{F} := (\Phi^1, \dots, \Phi^I) = \begin{pmatrix} \phi_1^1 & \cdots & \phi_1^I \\ \vdots & \ddots & \vdots \\ \phi_D^1 & \cdots & \phi_D^I \end{pmatrix} \quad (3.31)$$

Typical examples for supervised learning are classification scenarios. In this case, we want to assign a class label²⁰ $z(i) \in [1 : Z]$ to each instance $i \in [1 : I]$ of a dataset. With $Z = 2$, we speak of a **two-class problem** (binary classification). Scenarios with $Z > 2$ are **multi-class problems**. The classification algorithm or **classifier** learns a model for the classes using a set of training data with corresponding class labels. According to the learned model, the classifier predicts the classes for a test set consisting of unlabeled examples. The fraction of correctly classified test examples (**accuracy**) may serve as a metric to quantify the classifier’s performance. For a multi-class problem, it can be useful to calculate the mean accuracy over all classes. Nevertheless, this single number does not necessarily reflect properly the characteristics of a classification result [233]. We obtain more information by looking at confusion matrices or the stability of classification when changing parameters or experimental configurations.

For optimally exploiting the available data, we apply a procedure called **cross validation** (CV). Thereby, we split the data in $Y \in \mathbb{N}$ **folds**. One of the folds serves as test data,

²⁰There are also strategies for multi-label classification. We do not consider such approaches in this thesis.

	Fold 1	Fold 2	Fold 3
Round 1	Training fold	Training fold	Test fold
Round 2	Training fold	Test fold	Training fold
Round 3	Test fold	Training fold	Training fold

Figure 3.18. Three-fold cross validation. Each of the data folds serves as test data in one round.

the remaining folds as training data. We run this for Y rounds—once using each fold as test set—and calculate the average accuracy over all runs (Figure 3.18). An extreme case is **Leave-One-Out** CV where the test set only comprises a single instance ($Y = I$). In general, we have to make sure that the class distribution in the training set equals the overall distribution (**stratified** CV). Usually, the partitioning of instances into CV folds is a randomized process. For this reason, it may be useful to perform several runs of the whole CV procedure with re-initialized folds in order to analyze the stability of the classification results with respect to the fold partitioning.

3.6.2 Clustering

3.6.2.1 K-Means Clustering

For unlabeled data, an automatic (unsupervised) clustering of instances constitutes a useful analysis since it can reveal inherent structures of the data. A cluster comprises instances that are close to each other in the feature space—according to a suitable metric such as the Euclidean distance. The most common algorithm in this field is ***K*-Means Clustering** [140,144]. As the general idea of this method, we iteratively refine the assignment of instances to a cluster until the cluster centroids are stable (local optimum). Hereby, we assume that the instances in a cluster have a spherical distribution.

In *K*-means clustering, the number of clusters $K_{KM} \in \mathbb{N}$ is an important parameter since the quality of the clustering result crucially depends on K_{KM} . Scholars proposed several methods to automatically determine the optimal value for K_{KM} . In Chapter 7, we make use of the **silhouette score**, which quantifies the similarity of the instances within a cluster [207].

3.6.2.2 Hierarchical Clustering

The design of the *K*-means algorithm allows to express only one layer of clusters. For many applications, a hierarchical structure turned out to better represent the similarities of the data (**hierarchical clustering**). Typically, such structures consist of specific clusters and more general **cluster families**. In the field of bioinformatics, there are numerous methods for applications such as clustering of DNA sequences. One example are **Phylogenetic Trees**, which serve to represent evolutionary relationships as branching diagrams [88]. For a computational construction of such trees, a number of techniques exist. A simple bottom-up method is **neighbor-joining**, which bases on multiple sequence alignment. More advanced approaches consider evolutionary models such as the **minimum-evolution principle** [54]. In Chapter 7, we compute such phylogenetic trees to hierarchically cluster pieces by different composers.

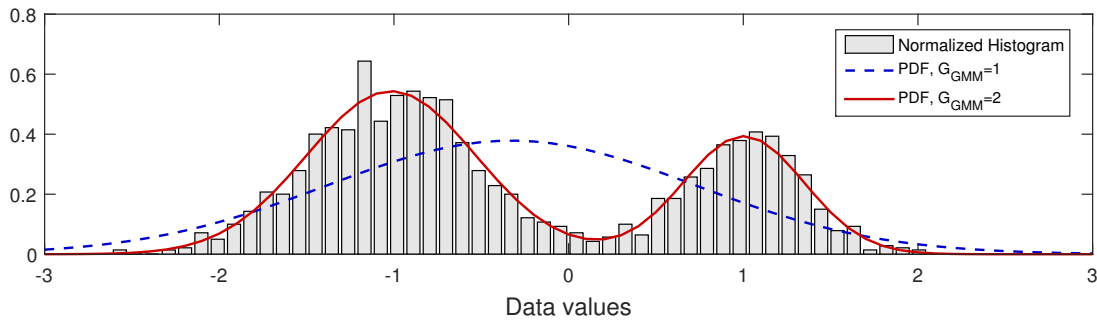


Figure 3.19. Gaussian Mixture Model. The histogram indicates the distribution of the data values. Here, $G_{\text{GMM}} = 2$ Gaussians seem to be necessary to resolve the structure of the data.

3.6.3 Classification

Classification is a supervised learning method. For well-defined scenarios, the classifier should learn to discriminate the classes in a *robust* way and be capable of adapting to unseen data—denoted as **generalization** [5]. To obtain good generalization, we have to consider the size and variety of the training data. Moreover, the model complexity plays a role. Complex classifiers usually comprise a lot of free input parameters. A small training set in combination with a complex classifier may lead to an over-adaptation—or **overfitting**—to the training data, which results in bad generalization. For example, complex models together with a large feature dimensionality D may cause this effect. The latter problem is known as the **curse of dimensionality** [20, 246]. As a rule of thumb, scholars consider a number of $\geq 10 \cdot D$ training instances per class as sufficient to prevent overfitting [107, 198]. For larger feature vectors, dimensionality reduction is necessary. A popular way to do this is **feature space transformation**, which we introduce in Section 3.6.4. Beyond that, **feature selection** can be a helpful strategy where we can additionally gain some insight into the relative importance of the feature dimensions. One example is **Inertia Ratio Maximization Using Feature Space Projection** (IRMFSP) proposed by Peeters and Rodet [187].

Scholars divide classification algorithms into two groups. **Generative classifiers** make use of probabilistic models and estimate the model parameters from the training data. **Discriminative classifiers** derive optimal decision boundaries from the training data [37, 172].

3.6.3.1 K-Nearest-Neighbor Classifier

A simple discriminative model is the K **Nearest Neighbor** ($K\text{NN}$) classifier. Based on a suitable distance measure, we consider the K_{KNN} training instances having minimal distance to a test instance and assign a class label with a majority decision [5]. The parameter $K_{\text{KNN}} \in \mathbb{N}$ controls the classifier’s sensitivity against outliers and local fluctuations in the feature space.

3.6.3.2 Gaussian Mixture Model Classifier

The **Gaussian Mixture Model** (GMM) is a generative classifier that estimates probability density functions in the feature space for each class (derived from the training data). These distributions are weighted sums of $G_{\text{GMM}} \in \mathbb{N}$ multivariate normal distributions. For each class, we estimate the parameters (mean vectors, covariance matrices, and weights) by maximizing the likelihood on the training data. An efficient strategy for this step is

the expectation-maximization algorithm [53]. With the parameter G_{GMM} (the number of Gaussians), we can control the model complexity (see Figure 3.19).

3.6.3.3 Support Vector Machine

Regarding discriminative classifiers, a widely used method is the **Support Vector Machine** (SVM) introduced by Vapnik *et al.* [44,243]. This algorithm aims at finding a hyperplane that optimally separates the classes. Among several possibilities, we choose the hyperplane that maximizes the distance between the separating plane and the closest data points (**maximum margin classifier**). Often, a hyperplane that perfectly separates the training instances does not exist. For this non-separable case, Cortes and Vapnik [44] proposed a solution using **slack variables** to minimize the general error for a non-perfect hyperplane (soft margin hyperplane). We can control these variables with an error penalty parameter C_{SVM} .

The basic SVM algorithms works with a linear hyperplane in a space of dimensionality $(D-1)$. In many scenarios, we may achieve better class separation with a nonlinear classifier. We obtain a nonlinear classification by applying the **kernel trick** [4,24]. Using a nonlinear mapping function, we fit a linear hyperplane in a higher dimensional space. As the basic idea of the kernel trick, we do not transform all the data points but directly compute the scalar product in the new space from the initial coordinates by using a **kernel function**. In our experiments, we use the **Radial Basis Function** (RBF) kernel. In this case, the performance of the SVM depends on the parameters C_{SVM} and γ_{SVM} . To optimize these parameters for the specific problem, Hsu *et al.* suggest a multistage **grid search** by using an internal 5-fold cross validation on the training set [97]. This step usually improves performance but makes the training of an SVM with RBF kernel a time-consuming procedure. In its original form, the SVM is a binary classifier. In order to apply this algorithm to multiclass problems, we split the task into several binary problems. Hereby, we use a **one-versus-one** strategy as implemented in the LIBSVM library [32].

3.6.3.4 Random Forest Classifier

Another discriminative method is the **Random Forest** (RF) classifier [6,25]. This algorithm makes use of the **ensemble learning** strategy based on decision trees. A decision tree is a hierarchical, rule-based model composed of internal **decision nodes** and terminal **leaves** [5]. Each node constitutes a discriminant value—or decision boundary—for one or more feature values with discrete outcomes labeling the branches. The leaf nodes assign the output values.

A basic tree construction procedure may lead to an over-adaptation of the trees to the training data. To overcome this problem, one possibility is to reduce the complexity (size) of the trees by removing irrelevant branches (“pruning”). As another strategy, the RF classifier makes use of **bootstrap aggregation** or **bagging** by selecting a random subset of instances for several training steps. The number of trees B_{RF} is a parameter. Additionally, the RF employs a **feature bagging** approach by only using a subset of the features for training each node (**random subspace method**). In a large study comprising different machine learning tasks, RF classifiers performed similar to SVMs [62]. In comparison, the training of RF is less time-consuming.

3.6.3.5 Further Classification Algorithms

Beyond the presented algorithms, many more classification methods were proposed. For image classification problems, **Sparse Representation Classifiers** turned out useful [262].

The basic idea of this strategy is to model the feature vector of a test instance as a linear combination of training feature vectors. Hereby, the algorithm prefers **sparse** linear combinations in the sense that only few of the training instances have non-zero coefficients.²¹

Furthermore, **Artificial Neural Networks** showed good performance on machine learning tasks [5, 152]. Neural networks consist of several **layers** of nodes. Beyond the input layer (the features) and the output layer (the class labels), **hidden layers** serve to connect input and output using complex non-linear combinations of the previous layers. Recently, **Deep Learning** techniques have become popular. These methods use a high number of hidden layers and can apply techniques for automatic learning of features. Due to their high complexity, these models are highly sensitive to overfitting when dealing with small or unbalanced training datasets. Additionally, it is hard to get an insight into their semantic behavior. Due to these reasons, we do not use such algorithms in this thesis.

3.6.4 Dimensionality Reduction

3.6.4.1 Principal Component Analysis

In machine learning problems, the number of features can be quite large ($D \gg 100$). Often, the feature matrix shows some kind of redundancy so that a lower dimensionality may be sufficient to capture the relevant information. In this case, **dimensionality reduction** techniques can be useful in order to obtain a representation of lower dimensionality $L < D$. One of the unsupervised methods to do this is **Principal Component Analysis** (PCA). This method constitutes a transformation of the feature vectors into a new basis with orthonormal basis vectors $\mathbf{w}^l := (w_1^l, \dots, w_D^l)^T \in \mathbb{R}^D$, $l \in [1 : D]$. The entries of \mathbf{w}^l are called weights or **loadings**. The first component \mathbf{w}^1 points towards the **maximum variance** direction of the feature space. With increasing index l , a vector \mathbf{w}^l describes a smaller fraction of the data's variance. Therefore, we can reduce the dimensionality of the feature space by only keeping the first $L < D$ components while still describing a large part of the variance.

We can express the feature vectors in the new basis as

$$\Phi^i = \sum_{l=1}^D \lambda_{i,l} \mathbf{w}^l \quad (3.32)$$

with principal component **scores** $\lambda_{i,l} = (\mathbf{w}^l)^T \Phi^i$. As an important preprocessing step for PCA, we have to subtract the mean vector over all instances from the initial feature vectors. Furthermore, it can be useful to divide the feature values by the standard deviation over all instances in order to equalize the contribution of the feature dimensions [5].

3.6.4.2 Linear Discriminant Analysis

For scenarios with multiple classes, we might additionally take into account the class labels for dimensionality reduction. One example of such supervised methods is **Linear Discriminant Analysis** (LDA) [5, 146, 248]. Hereby, we want to optimally separate the instances belonging to different classes. We therefore try to find a representation that *maximizes* the variance *between* different classes while *minimizing* the variance *within* each class (**Fisher's**

²¹Sturm and Noorzad [234] showed that applying such a classifier to audio-based music genre recognition—in conjunction with timbre-related features—may capture irrelevant properties that only correlate to genre in a specific dataset. Similar observations were made for other classifiers, too.

criterion). Researchers showed that for Z classes at most $(Z - 1)$ linear independent dimensions exist [5]. In comparison, using LDA as feature space transformation often leads to better classification performance than using PCA. However, exceptions from this behavior occur when the training data does not properly represent the underlying statistical distribution of the feature space [146].

3.6.4.3 Further Dimensionality Reduction Techniques

The methods presented above consider the whole feature space for calculating the transformation. For this reason, they are sensitive to outliers. A more sophisticated method are **Self-Organizing Maps**. This reduction technique considers the local neighborhood of an instance by means of pairwise distances in order to preserve the topological characteristics of the initial feature space [120]. A related method is **Multidimensional Scaling** [125]. In contrast to self-organizing maps or LDA, this technique only requires as input the distances between the instances and not the full feature vectors. In this thesis, we do not use these more sophisticated methods for dimensionality reduction.

4 State-of-the-Art

In this chapter, we give an overview of algorithms for tonality analysis and style classification published in the area of Music Information Retrieval (MIR). Concerning the representations of music, we focus on methods for audio data but also mention important work treating symbolic data. Furthermore, we summarize publications dealing with methods for analyzing and organizing music datasets and archives (classification and clustering).

4.1 Overview

First, we want to give an overview of the most important tasks in this research area. To this end, we present in Figure 4.1 a **specificity-granularity plane** inspired by Grosche *et al.* [81]. We focus on the musical parameters *tonality* and *style* and ignore other aspects that principally might be relevant for our work.¹ The most detailed analysis of audio recordings is automatic transcription. Despite considerable progress in recent years, this task still remains challenging for many scenarios. Complex instrumentations with subtle timbral differences—such as a Romantic symphony orchestra—or polyphonic textures with many musical voices provide major problems for transcription algorithms. However, scholars successfully approached secondary concepts such as chords or key without having perfect transcription systems. These methods typically rely on chroma features for capturing relevant pitch class information.

As shown in Figure 4.2, different types of tonal structures hierarchically depend on each other. In particular, there is no agreement among musicologists which layer (chords or scales) is the more fundamental one (as discussed in Section 2.7). In the next sections, we approach these concepts in the following order:

- **Global key detection** (Section 4.2) is a straightforward task where we want to assign a single key label (tonic note and mode) for the whole movement.
- **Local key analysis** (Section 4.3) attempts to resolve key changes (modulations) that occur throughout a movement. Here, scholars either try to partition a piece into key segments [31, 267] or propose visualizations that account for ambiguities [209, 210].
- **Chord recognition** (Section 4.4) refers to a finer temporal level. Here, the aim is to find appropriate chord labels together with the corresponding start and ending time.

Some methods concurrently approach two or more of these layers [148, 155, 180, 201, 222]. For segmentation, the latter two applications employ either rule-based strategies or dynamic programming—often using Hidden Markov Models (HMMs). Besides the 24-key problem, several researchers propose different systems for local tonality analysis relying on—among others—diatonic scales [266]. In Section 4.5, we summarize more abstract concepts for describing tonality such as **tonal complexity**, harmonic tension, or degrees of tonality. Based

¹For example, some authors combine beat tracking with tonality analysis in order to obtain musically relevant segments [148, 180].

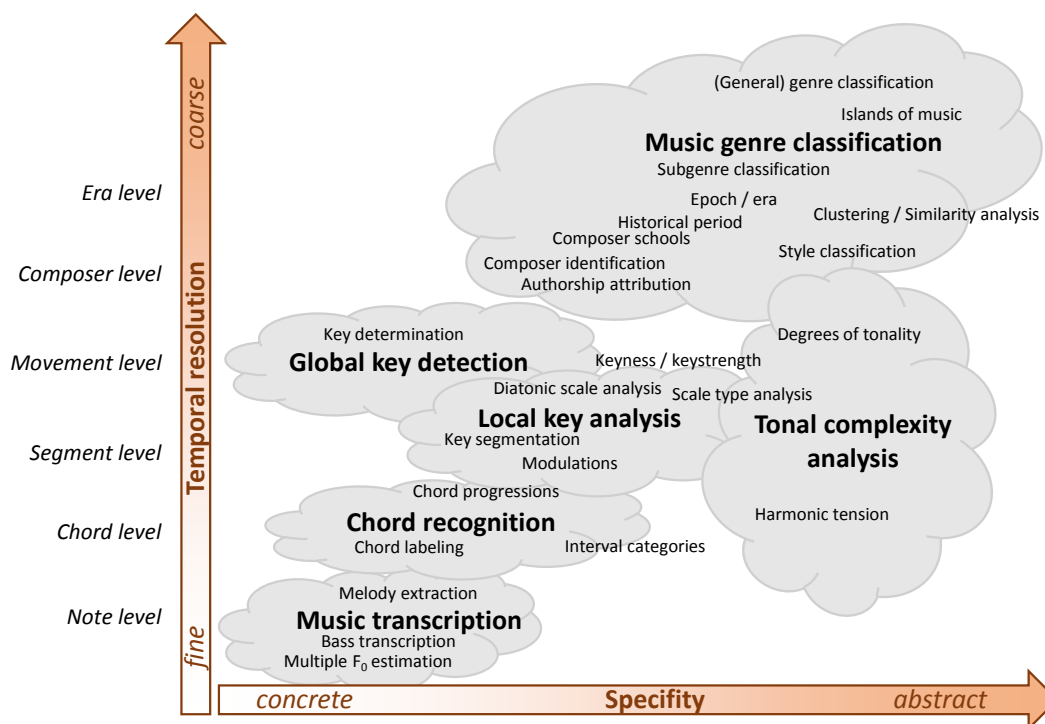


Figure 4.1. Overview of tonality and style analysis tasks. This figure visualizes different topics from the field of MIR research. Inspired by [81], we arrange the tasks according to their semantic relation in a specificity-granularity plane. The vertical axis indicates the temporal scale the concepts refer to. The specificity of the concepts is given by their horizontal position. Here, we restrict ourselves to methods for tonality and style analysis and ignore other musical parameters such as rhythm, meter, melody/motifs, or instrumentation.

on such descriptors, some authors classify music into tonal and atonal pieces [93]. More concrete experiments address the automatic categorization into historical periods or try to identify the composer of a piece. In MIR research, this is considered as a specific case of **music genre classification**—with style-related subgenres of the top-level genre “classical.” In Section 4.6, we present a detailed summary of the work in this field concerning both symbolic and audio data.

For the majority of applications, researchers mainly focus on popular music. For example, many authors evaluate their chord recognition systems on songs by The Beatles and other pop music. Because of this, we mention the most important contributions from this field even though we are mainly interested in studies performed on Western art music.

In general, a quantitative comparison of results is problematic. Even though many publications deal with similar or identical tasks, the experimental settings—such as the size and the structure of datasets, the number and the definition of classes, or the chord types considered—as well as the evaluation measures vary widely. For learning-based approaches, the experimental design with respect to training and evaluation (cross validation) may exhibit crucial differences. Nevertheless, we try to mention the central results of the publications together with the important details of the evaluation procedure. It is important to be very careful with a direct comparison of these numbers.

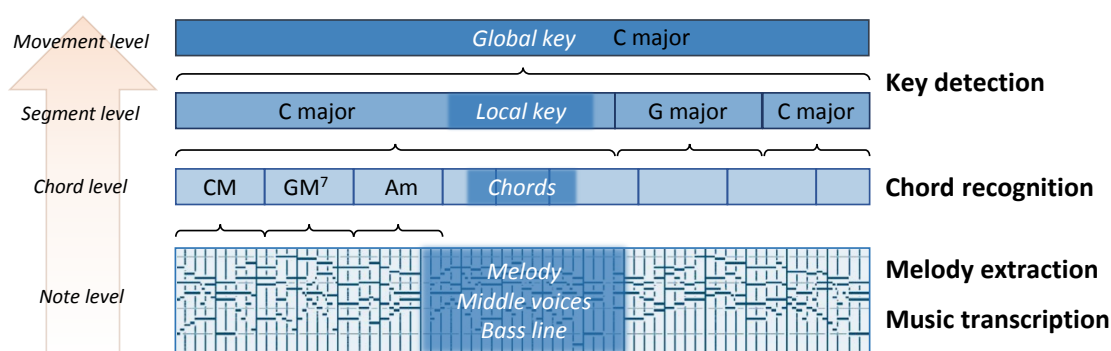


Figure 4.2. Hierarchical nature of tonal structures. This overview visualizes different concepts for tonality analysis of a movement with respect to their temporal levels.

4.2 Global Key Detection

This overview follows the one given in [252]. Since it is a standard task in MIR research, a number of scholars proposed methods for detecting the global key (see Section 2.7) from audio. There are algorithms dealing with symbolic data only as well as audio analysis methods, on which we focus on here. Several authors give overviews of the state-of-the-art [180, 218] and compare **knowledge-based** and **data-driven** algorithms—the two main approaches. The latter category (data-driven) usually requires a training stage whereas strategies from the other domain are rule- and parameter-based.

In general, the first step is an extraction of chroma features. Motivated by studies on human pitch perception [124, 235], many algorithms match the chroma statistics to pitch class templates or use advancements of such approaches [75, 78, 103, 180, 183, 222]. For example, Zhu *et al.* [268, 269] introduce a tuning estimation stage as well as overtone reduction (consonance filtering) and temporal smoothing for improving chroma robustness. They use a two-step approach for scale and tonic note estimation and obtain success rates of 81 % for 72 orchestral pieces from different eras [268].

Izmirli [104] shows that reducing the feature dimensionality from twelve to not less than six dimensions (using PCA) does not deteriorate key detection performance. He evaluates on 152 classical pieces from commercial recordings. Shenoy *et al.* [222] first conduct a simple chord detection algorithm and estimate the key by comparing chord statistics. They obtain 90 % accuracy for 20 pop songs.

Among the systems using data-driven techniques, HMMs constitute a popular method [31, 174, 186]. Chai and Vercoe [31] combine HMMs with a two-step approach, considering the key signature (the most prominent diatonic scale) and the tonic note individually. Noland and Sandler [174] investigate the effect of signal processing parameters and test their HMM-based approach on recordings of Bach’s well-tempered piano, first book (48 tracks), yielding 98 % correct classification for the best parameter settings. Peeters [186] trains HMM models for the 24 keys and evaluates on 302 classical pieces from different eras with 86 % accuracy. He obtains an improved result of 89 % when using an overtone reduction algorithm (harmonic peak subtraction) for computing the chroma features [185].

There are several works considering special sections of the recordings. Izmirli [103] investigates the first seconds of 85 classical pieces by different composers with up to 86 % success. Chuan and Chew [39] test their geometrical approach on the beginning of several Mozart symphonies yielding up to 96 % success rate. Extending these experiments to a wide stylistic range [38], they reach 75 % correct keys. Mehnert *et al.* [157] propose another spatial pitch

model (symmetry model) for key detection and evaluate on a mixed dataset with 83 % correct keys. Van de Par *et al.* [239] combine profile training with a special weighting of the beginning and ending section (15 seconds). They evaluate on piano music² with high accuracies up to 98 %. In the MIREX contest³ (1252 classical pieces synthesized from MIDI), the best results of the past years reached 87 % correctly identified keys [102].

In conclusion, efficient detection of the global key is possible with a number of different strategies. However, exceeding the glass ceiling of about 90 % accuracy—without overfitting to a specific dataset or musical style—seems to remain challenging.

4.3 Local Key and Modulations

As we mentioned in Section 2.7, the musical key may change over the course of a piece. To account for such changes (modulations), several researchers extended key analysis to a local approach. For this task, the annotation and evaluation is time-consuming and often not consistent among different annotators and task definitions. This makes a comparison of the algorithms problematic.

Izmirli [105] combines local key finding with non-negative matrix factorization (NMF) for segmentation. A number of scholars considered HMMs for this task—such as Chai and Vercoe [31] who compared the 12-key problem (without mode detection) to the classification of 24 keys. Zhu and Kankanhalli [266] propose diatonic scale estimation for addressing the global 12-key problem and further apply this model to key-based melody segmentation in pop songs [267]. They test their approach on a small dataset of monophonic MIDI signals.

Several methods address the problem of chord detection and local tonality at the same time [148,180,201]. Often, beat tracking serves as a preprocessing step in order to obtain musically meaningful analysis windows [148,180]. Papadopoulos and Peeters [180] simultaneously treat global and local key finding by incorporating downbeat information. They evaluate their system on two datasets of different styles (Mozart piano sonatas and pop songs) with key detection accuracies up to 80 %.

Compared to popular music, we find less contributions regarding local tonality in classical music. One reason for this may be the ambiguous nature of segment borders in classical music. The modulation types described in Section 2.7 usually proceed gradually over a certain time span. In contrast, pop music often employs abrupt or fast changes using only few pivot chords. Mearns *et al.* [155] try to detect modulations in synthesized recordings of twelve chorales by J. S. Bach. As a first step, they perform automatic transcription. On the transcribed music as well as for reference MIDI data, they recognize the chords and finally estimate the local key segmentation from the chord progressions using HMMs based on music theory models. Though transcription performance is low, they obtain good local key detection results for both audio- and MIDI-based segmentation.

Because of the ambiguous and time-consuming annotation procedure, several researchers restrict themselves to a visualization of local keys in classical music rather than segmenting and evaluating quantitatively. Purwins *et al.* [194] use a very basic approach for local key tracking by extending the template-matching approach to local windows. They obtain interesting results for a piano *prélude* by F. Chopin. For such visualizations, the time resolution of the local windowing plays a crucial role. Sapp [209,210] proposes a useful technique for visualizing several time scales simultaneously by using **scape plots** for local key analysis.

²See Section 5.1 and [183] for detailed information about this dataset.

³<http://www.music-ir.org/mirex>

Jiang and Müller [110, 168] adapt this method for structural and tonal analysis of piano sonatas by L. van Beethoven.

4.4 Recognition of Chords and Chord Progressions

Since chords constitute an important concept for composing, playing, and analyzing Western music, numerous publications deal with the automatic extraction of chord symbols from audio. Fujishima [67] first proposed a system for estimating several chord types over time using chroma features. In [221], Sheh and Ellis introduce HMMs with Viterbi decoding in order to perform smoothing and chord estimation at the same time. Many researchers experimented with improvement strategies to this fundamental approach. As the main ideas, they try to enhance the robustness of chroma features [57, 131, 147, 238] or introduce complex chord models [29, 132, 260]. Some methods incorporate beat tracking as a preprocessing step [21, 57] or concurrently estimate several idioms such as downbeat, chords, and key [148, 201]. For a detailed overview of contributions to the chord recognition task, we refer to the comprehensive study by Cho and Bello [36].

Most of the studies presented above evaluate the proposed algorithms on popular music. A major issue of chord recognition methods is the selection of possible chord types. A large fraction of the approaches only use major and minor triads, which cannot fully describe the harmonic content—even for popular music. On the other side, the use of a large chord dictionary deteriorates robustness of chord detection and may lead to rather artificial chord changes such as $CM - CM^7 - CM$ or similar progressions, which do not describe the musical content in a meaningful way. Furthermore, signal processing artifacts such as the influence of overtones may lead to confusions for difficult chords [147]. The selection of meaningful chord types considerably depends on the musical style. In jazz or Romantic music, in particular, complex chord types may arise. Konz *et al.* [59, 121] performed several studies to evaluate the consistency of chord recognizers. In [121], they systematically compare the results of such an algorithm for different interpretations of the same piece (L. van Beethoven’s “Appassionata”).

Several researchers automatically analyzed chord progressions. Inspired by language models, they most often describe these progressions as probabilistic n -grams and analyze their statistics for music databases [33, 175, 188, 213, 265]. Scholz *et al.* [213] perform such a study on manually labeled chord sequences of Beatles songs and show the efficiency of smoothing and selection techniques. Using the same data, Yoshii and Goto [265] extend this approach with a nonparametric Bayesian model. Mauch *et al.* [149] analyze manually labeled chord progressions from 400 Beatles songs and jazz standards. From the same data, Anglade and Dixon [9] automatically derive harmony rules using inductive logic programming. Concerning classical music, Kaneko *et al.* [112] analyze 50 manually labeled pieces using chord bigrams.

Beyond such manually extracted chord labels, a number of researchers proposed methods for obtaining harmony rules or harmonic grammar from symbolic data [15, 47–50, 101, 177, 203]. Paiement *et al.* [177] suggest to use tree structures instead of HMMs and evaluate on jazz standard themes in a MIDI representation. Barthélemy and Bonadi [15] try to extract the harmonic content in form of a figured bass using automatic score reduction. Others use complex hierarchical models to describe chord progressions within a larger tonal framework [49, 203].

A number of authors used automatic chord recognition algorithms as basis for analyzing chord progressions from audio. Cheng *et al.* [33] use an n -gram model in an HMM framework to derive chord progression probabilities. They obtain best results for $n = 3$ and $n = 4$ based

on 28 Beatles songs. Mauch *et al.* [151] perform a large analysis of chord progressions in the US pop music charts using the Chordino Vamp plugin.⁴ With the same software, Barthelet *et al.* [16] summarize chord bigram probabilities for several musical styles from a commercial audio collection including nearly 27 000 classical music tracks. They provide a web interface⁵ for exploring the extracted chord progressions by means of different visualizations [111].

4.5 Tonal Complexity

Apart from concrete tonal items, several researchers introduce methods for measuring more abstract concepts such as tonal complexity. We discussed the musical implications of this notion in Section 2.9. Concerning the computational analysis of such concepts, Parry [181] analyzes the complexity of popular music but focuses on rhythmic and melodic aspects. Honingh and Bod [92] evaluate the suitability of pitch class set categories for measuring degrees of tonality based on MIDI data. Analyzing classical pieces from different composers, they found an interesting correlation between the presence of interval category IC5 (P4 and P5 intervals, see Table 2.3) with a decrease of tonal complexity. For the purposes of style classification and authorship analysis based on symbolic data, Kranenburg *et al.* [12,241,242] make use of entropy measures for pitches, chords, and sonorities.

Streich and Herrera [230,231] discuss harmonic complexity as one facet of overall music complexity and propose an audio-based method for describing this notion. They measure the relation between the local tonal content in a short-time window to the one in a longer window. With a similar approach, Mauch and Levy [150] analyze and visualize the structural change of musical pieces based on—among others—tonal complexity.

4.6 Classification and Clustering

4.6.1 Overview

In MIR, the classification of audio data into genres and stylistic categories constitutes a central task [46,237]. For an overview of this field, we refer to the article by Weihs *et al.* [251]. In Figure 4.3, we illustrate the hierarchical nature of genre classification tasks. The majority of studies focus on top-level genres such as Rock, Pop, Jazz, or Classical. There are several attempts to obtain a finer class resolution by considering sub-classes of individual genres such as Rock [236], Electronic [70], or Ballroom dance music [55]. Further studies consider global cultural areas as subgenres [126,179]. For approaching these tasks, most methods make use of timbral or rhythmic features. In contrast, there are only few methods concerning the subgenre classification of classical music. In this section, we give an overview of studies for clustering and classification of both composers and stylistic periods. We focus on methods that use features for describing tonal aspects of the music. Table 4.1 gives a summary of the most important contributions. To get a rough overview of the methods' performance, we list the classification accuracies as reported in the respective publications. However, a comparison of these values is very problematic since the experimental configurations vary widely.

⁴<http://isophonics.net/npls-chroma>

⁵<http://dml.city.ac.uk/chordseqvis>

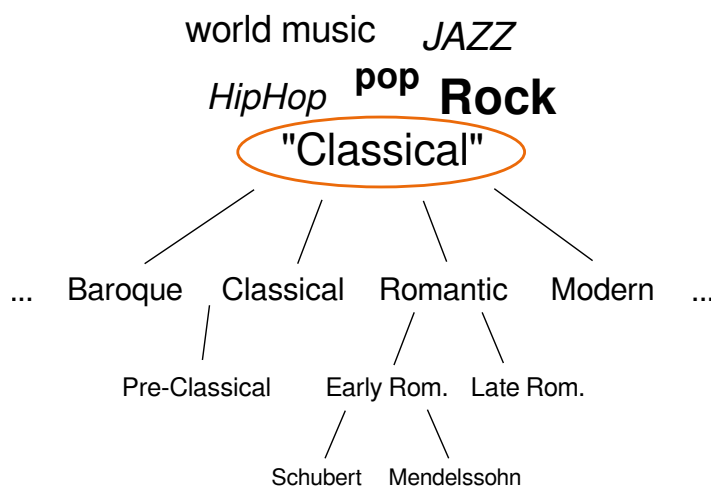


Figure 4.3. Different levels of music genre classification. The labels in the upper part refer to the top-level genres. Regarding classical music, typical subgenres are historical or stylistic periods (middle layers). Composer identification (lower part) constitutes a more specific task.

4.6.2 Studies on Symbolic Data

Concerning style classification of Western classical music, we find several studies based on scores or symbolic data. McKay and Fujinaga [153] perform hierarchical classification into root genres (classical, jazz, and pop) and leaf genres (three for each root genre) using high-level musical features extracted from MIDI data. As classical subgenres, they consider the periods Baroque, Romantic, and Modern with a success rate of about 85% within the full hierarchical classification. Ogihara and Li [175] analyze progressions from chord symbols for clustering and measuring similarity among eight jazz composers. Hedges *et al.* [89] extend this idea and perform classification experiments with multiple viewpoint Markov classifiers based on chord sequences. Among other tasks, they evaluate classification of nine jazz composers with 67% accuracy. Furthermore, they classify into eight jazz subgenres obtaining 58% accuracy in the optimal setting. De Leon and Iñesta [192] test different pattern recognition approaches for discriminating the genres jazz and classical. They calculate different measures from MIDI representations of monophonic melodies.

Regarding classical music, Geertzen and van Zaanen [73] estimate rhythmic and melodic structures from scores using grammatical inference. They obtain up to 80% accuracy for two- and three-composer classification tasks. Mearns *et al.* [156] perform classification experiments for seven composers from the Renaissance and Baroque periods. Based on score data, they calculate high-level features for quantifying harmonic intervals as well as properties of counterpoint obtaining 66% accuracy on a small dataset. Van Kranenburg *et al.* [12, 242] evaluate different composer identification and clustering tasks on score data using interval- and pitch-related features as style markers. For the five-composer problem (Bach, Handel, Telemann, Haydn, and Mozart), they obtain classification results of about 75%. However, they also test other class constellations such the “Bach-vs.-all” scenario with up to 95% accuracy. Moreover, they perform visualizations for studying works of uncertain authorship in a suitable feature space using LDA [12, 241]. Among other features, they also quantify notions such as pitch entropy, which exhibits some relation to tonal complexity. Hontanilla *et al.* [95] use the five-composer data from [242] and classify on the basis of language models (4-grams) for melodies. They obtain a similar result of 79% for the five-composer task.

Table 4.1. Clustering and classification experiments for musical styles. This overview summarizes relevant studies dealing with stylistic subgenres such as style periods or composers.

<i>Authors</i>	<i>Task</i>	<i>Classes</i>	<i>Repres.</i>	<i>Features</i>	<i>Acc.</i>
Symbolic data					
McKay & Fujinaga [153]	Classific.	3 classical styles	MIDI	various	85 %
Ogihara & Li [175]	Clustering	8 jazz composers	Chord symbols	Chord progressions	–
Hedges <i>et al.</i> [89]	Classific.	9 jazz composers	Chord symbols	Chord progressions	67 %
Hedges <i>et al.</i> [89]	Classific.	8 jazz styles	Chord symbols	Chord progressions	58 %
Mearns <i>et al.</i> [156]	Classific.	7 classical composers	Score	Intervals, counterpoint	66 %
Geertzen & van Zaanen [73]	Classific.	≤ 3 classical composers	Score	Melody & rhythm sequences	80 %
Ponce de León & Iñesta [193]	Classific.	2 styles (classical–jazz)	MIDI	Melody descriptors	90 %
Van Kranenburg & Backer [242]	Clustering & Classific.	≤ 5 classical composers	Score	Intervals, pitch entropy, counterpoint	80 %
Hontanilla <i>et al.</i> [95]	Classific.	≤ 5 classical composers	Score	Melody n -grams	79 %
Ventura [245]	Classific.	3 classical styles	Score	Melody entropy	–
Rodrigue Zivic <i>et al.</i> [202]	Clustering	historical periods	Score	Melodic intervals	–
Honingh & Bod [93]	Classific.	2 classical styles (tonal–atonal)	MIDI	Interval categories	95 %
Hillewaere <i>et al.</i> [91]	Classific.	2 classical composers (Mozart–Haydn)	MIDI	Melody n -grams and statistics	75 %
Dor & Reich [56]	Classific.	≤ 9 classical composers	Score	Pitch class, octave, melodic	79 %
Audio data					
Pérez-Sancho <i>et al.</i> [189]	Classific.	3×3 subgenres	Audio	Chord n -grams	68 %
Jiang <i>et al.</i> [108]	Classific.	5 (sub-)genres	Audio	MFCC, OSC	82 %
Hu <i>et al.</i> [98]	Classific.	9 classical composers	Audio	MFCC-like	76 %
Purwins <i>et al.</i> [195]	Clustering & Classific.	6 classical composers (“one-vs.-all”)	Audio	Chroma histograms, tonic-note-related	97 % ROC
Izmirlı [106]	Classific.	2 classical styles (tonal–atonal)	Audio	Chroma histograms	91 %
Hamel [85]	Classific.	11 composers (2011 MIREX task)	Audio	MFCC-like	78 %

Regarding tonal complexity, Perttu [190] studies the increase of chromaticism in Western music from the year 1600 to 1900 on score representations of musical themes. Ventura [245] uses score representations to identify the periods Baroque, Romantic, and Contemporary based on some kind of melodic entropy. He directly compares individual feature values for a small set of examples. As an early contribution, Fucks and Lauter [66] present statistical analyses of instrumental parts (violin, flute, and vocal) for about 100 examples. Among other

features, they compute kurtosis and correlation measures for distributions and transition matrices of pitches, note durations, and intervals. As their main finding, they measure a fundamentally different tonal behavior of atonal and tonal music. For such kind of melody-based studies, Viro’s “Peachnote” corpus [247] provides interesting material. This dataset contains statistics of melodic intervals obtained via optical music recognition from open-access graphical scores.⁶ On that data, Rodriguez Zivic *et al.* [202] perform unsupervised clustering experiments obtaining a division into the eras Baroque, Classical, Romantic, and Modern. The approach by Honingh and Bod [92, 93] relies on quantifying interval categories. They evaluate several clustering and classification tasks on MIDI representations of individual pieces. Among other experiments, they perform tonal-vs.-atonal classification with up to 95 % success rate [93].

Kiernan [116] tests classification of flute compositions by three composers using key-related pitch class occurrences from scores. After training, he investigates the system’s output on compositions with unknown authorship and, thus, does not report quantitative results. For the specific two-composer task of discriminating Mozart and Haydn string quartets, Hillewaere *et al.* [91] propose a MIDI-based approach. They calculate global features and estimate melody n -gram models to the individual parts of the string quartet. They achieve 75 % classification accuracy—with global features performing best on violin I parts, and n -grams being superior on cello parts. Dor and Reich [56] perform a large study by evaluating score-based features on several composer identification tasks. The dataset comprises piano pieces and works for strings. From a total of 1183 scores by nine different composers, they compile several subsets. Their feature set encompasses both absolute pitch class and octave statistics as well as note counts, durations, and melodic sequences (trigrams). With an automatic feature learning procedure, they evaluate the individual features’ impact. Hereby, pitch classes and octaves show high importance whereas adding melodic properties only leads to small improvements. For their two-composer experiments, they obtain accuracies ≥ 90 % except for Haydn–Mozart (63 %), Beethoven–Chopin (84 %), and Corelli–Vivaldi (85 %). In the instrument-specific experiments, the cases of string data yield slightly better results. In general, comparing scores for a specific instrumentation only shows higher recognition rates in most class constellations. For their maximal task of classifying eight composers, they obtain 79 % accuracy. Overall, absolute pitch class histograms show high impact in their experiments (≥ 60 % average contribution to two-composer results) even if they are not independent from the key of a piece. To the author’s knowledge, this comprehensive study [56] constitutes the state-of-the-art for composer identification based on symbolic data.

4.6.3 Studies on Audio Data

For classifying audio data, only few studies consider subgenres of classical music. Some of them use instrument categories as sub-classes [225]. By using a transcription system, Lidy *et al.* [138] adapt features for symbolic data from [193] combined with audio features for genre classification. Anglade *et al.* [8] follow a similar idea by using a chord detection algorithm. For the training, they learn harmony rules from symbolic data. In another genre classification study, Pérez-Sancho *et al.* [189] adapt their symbolic data approach based on chord n -grams [188] to the audio domain by using automatic chord transcription. They classify into three genres (including classical) with each three subgenres obtaining 68 % classification accuracy.

⁶<http://www.ims1p.org>

Jiang *et al.* [108] also use classical subgenres (Baroque and Romantic) together with other top-level genres. They obtain results of 82 % by using MFCC and OSC features.

For composer identification, Hu *et al.* [98] test an approach involving deep neural networks with MFCC-like features. They yield 76 % classification accuracy for a nine-composer task with about 360 clips of 30 s length per composer. Their dataset comprises pieces with several types of instrumentation. Purwins *et al.* [195] perform different ML experiments on a set of piano recordings from six classical composers. Their experiments rely on constant-Q chroma features summarized to global histograms. They obtain relative pitch class histograms by shifting the chroma histograms to the tonic note of the annotated key. Classifying the composers in a “one-vs.-all” setting, they obtain results between 72 % (Scriabin) to 97 % (Hindemith) area under the curve (AUC) measure using receiver operating characteristic (ROC) as evaluation method.⁷ With unsupervised clustering (*K*-means), the main separation occurs between pieces in major and minor mode. Concerning some exceptional and borderline data points, the authors mention several musical reasons. Using self-organizing maps, they find different regions in the feature space for individual composers. Similarly, Kaneko *et al.* [112] perform PCA on chord transition bigrams obtaining clusters with composers of an era. Izmirli [106] performs classification of tonal-vs.-atonal music based on chroma histograms. He obtains a classification accuracy of 91 %.

In the MIREX contest, one sub-task of genre classification addresses classical composer identification from audio data. The corresponding dataset consists of 2772 audio excerpts of 30 s length by 11 different composers (252 clips per composer). The annotations include information about the albums. According to the website,⁸ album filtering is applied in the evaluation (compare Section 8.3.3). Most submissions to this task are intended to serve for genre classification tasks in general. Concerning the maximum classification accuracy, the approach by Hamel [85] in 2011 reached the best result obtained so far. This system relies on spectral features related to MFCCs (“Principal Mel-Spectrum Components”) and uses feature pooling with a neural network.

In summary, most studies for automatic style recognition deal with symbolic data. The features often rely on melodic properties, but also chord progressions and pitch class occurrences are typical. For audio data, scholars employed both spectral- and chroma-based features with success. The reported accuracies reach up to 78 % [85, 98] for classifying nine and eleven composers, respectively. Thus, composer identification based on audio and symbolic [56] data leads to roughly similar results. However, it is difficult to directly compare published results since evaluation measures, experimental setup, and the data to analyze are varying to a high degree. Thus, systematic experiments to compare classification algorithms for Western classical music are yet to be done.

⁷This evaluation procedure for binary classifiers considerably differs from the mean accuracy [61]. Hence, a direct comparison of these numbers is not meaningful.

⁸<http://www.music-ir.org/mirex>

5 Analysis Methods for Key and Scale Structures

The contributions of this thesis address the automatic analysis and classification of classical music audio recordings. In this chapter, we present several methods for extracting tonal content from audio data. For all of these algorithms, we rely on some type of chroma features and derive measures for estimating the occurrence of certain tonal structures. We discussed the limitations of such strategy in Section 3.5.6. For some algorithms, we provide quantitative analyses on both publicly available and specifically created datasets. For other ideas, we demonstrate the potential by means of visualizations. In Section 5.1, we treat the problem of global key finding in classical music and propose an approach relying on the final chord. Section 5.2 describes analysis methods for the local presence of diatonic scales and different scale types in general, which we demonstrate for several pieces.

5.1 Global Key Estimation Based on the Final Chord

5.1.1 Introduction

In Western classical music, the global key plays an essential part for a piece’s tonal characteristics (see Section 2.7). Many works already include the key in their title such as “Symphony in G major.” For several composers, certain keys exhibit a particular semantic meaning [11]. Beyond this, global key information is crucial to relate tonal structures (pitch classes, chords, local keys, etc.) to the tonic note in order to obtain key-independent features. In this section, we propose and evaluate an approach for global key extraction from audio recordings restricting ourselves to Western classical music from the common-practice period. Our rule-based method relies on chroma features. We put special emphasis on the final chord of the piece for estimating the tonic note. To determine the mode, we analyze a chroma histogram over the complete piece and estimate the underlying diatonic scale. For both steps, we apply a multiplicative procedure obtaining high robustness against errors. This approach helps to minimize the number of tonic note errors, which is important for subsequent tonal analyses.

This section relies on the publication [252]. Partly, the results stem from [259] and the associated bachelor’s thesis by Schaab [211]. We first present the design of our key detection algorithm (Section 5.1.2). Then, we outline the results of several studies on the basis of suitable audio datasets (Section 5.1.3). For the details of musicological terminology, we refer to Chapter 2. Section 4.2 summarizes related work concerning global key detection.

5.1.2 Proposed System

In the presented key detection system, we make use of the final chord’s significance in Western classical music applying a two-step approach. First, we separately estimate the final chord’s root note and the complete piece’s dominating diatonic scale. Finally, we combine these results obtaining the most probable key candidate consisting of the tonic note and the mode. Figure 5.1 shows an overview of the processing flow.

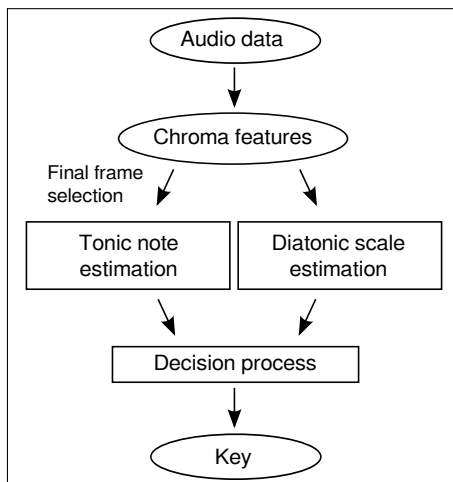


Figure 5.1. Overview of the key extraction process. After estimating the final chord’s root and the global diatonic scale, we combine this information in order to obtain the global key.

5.1.2.1 Feature Extraction

The algorithm presented in this section relies on chroma features. We use the Chroma Toolbox [165] to compute a pitchogram \mathcal{Y} in the piano range and derive CP chroma features, both with a temporal resolutions of 10 Hz. To account for the global tuning, we use the tuning estimation functionality of this toolbox package and apply a shifted filter bank as soon as the difference from a 440 Hz tuning exceeds 15 Cent. We obtain a log-frequency spectrogram (pitchogram) $\mathcal{Y}(p, m)$ with $p \in [21 : 108]$ and $m \in [0 : M - 1]$ (see Section 3.5). To estimate the overall energy, we calculate the average ℓ_1 norm (Equation (3.23)) of the pitchogram frames $\mathbf{p}^m := \mathcal{Y}(\cdot, m)$:

$$E_{\text{mean}} = \frac{1}{M} \sum_{m=0}^{M-1} \ell_1(\mathbf{p}^m) \quad (5.1)$$

Furthermore, we calculate a normalized chromagram \mathcal{C}^{ℓ_2} as well as a normalized chroma histogram \mathbf{g}^{ℓ_2} as presented in Section 3.5.5.

5.1.2.2 Tonic Note Estimation

On the basis of this feature set, we estimate the root note of the piece’s final chord. Since we do not want to consider frames containing silence, we take the last F feature frames that exceed a defined energy threshold. To account for the overall loudness of the piece, we apply a dynamical adaption for the energy threshold. To this end, we calculate the ℓ_1 norm for each of these pitch feature vectors \mathbf{p}^m and select only frames m that fulfill the condition

$$\ell_1(\mathbf{p}^m) > \rho \cdot E_{\text{mean}} \quad (5.2)$$

with a suitable factor $\rho \in \mathbb{R}^+$. From the frame selection thus obtained (length F), we compute a normalized chroma histogram $\mathbf{h} := (h_0, \dots, h_{11})^T$ similar to Equation (3.30) but using the Euclidean norm ℓ_2 here. To consider the tonal relationship between the chroma classes, we re-sort the entries of \mathbf{h} according to a series of perfect fifths by re-ordering the

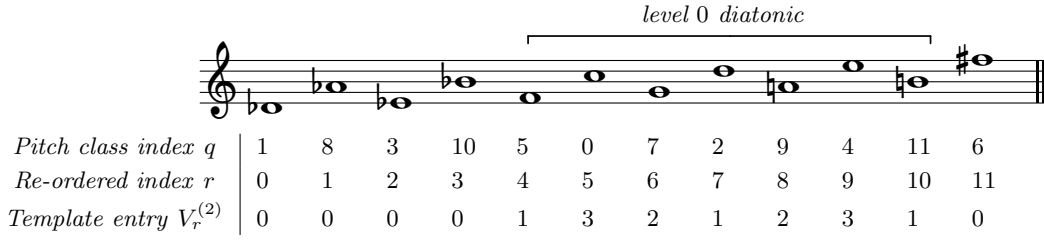


Figure 5.2. A diatonic subset (level 0) of the fifth-ordered chromatic scale. The first row indicates the pitch class indices from Equation (3.21) with $q = 0$ denoting C. The second row gives the re-ordered indices r . In this notation, the notes with indices $r \in [4 : 10]$ are forming the level 0 diatonic scale (C major scale). For this scale, we show in the third row the exponents for the specific template $\mathbf{V}^{(2)}$.

indices $q \rightarrow r := (q \cdot 7 + 5) \bmod 12$:

$$(0, 1, \dots, 11) \rightarrow (5, 0, 7, 2, 9, 4, 11, 6, 1, 8, 3, 10) \quad (5.3)$$

We obtain a fifth-ordered chroma histogram $\mathbf{h}^{\text{fifth}} := (h_0^{\text{fifth}}, \dots, h_{11}^{\text{fifth}})^T \in \mathbb{R}^{12}$. The indices $r \in [0 : 11]$ correspond to the pitch classes in the following way (Figure 5.2):

$$(0, 1, \dots, 11) \hat{=} (\text{Db}, \text{Ab}, \text{Eb}, \text{Bb}, \text{F}, \text{C}, \text{G}, \text{D}, \text{A}, \text{E}, \text{B}, \text{F}\sharp) \quad (5.4)$$

This pitch class ordering relates to the key arrangement in the circle of fifths (see Figure 2.16).

We now multiply these values for each two neighboring entries in order to consider only those chroma peaks that also contain some energy in the corresponding upper fifth chroma (Figure 5.3). This results in a product histogram $\mathbf{h}^{\text{prod}} := (h_0^{\text{prod}}, \dots, h_{11}^{\text{prod}})^T$ with

$$h_r^{\text{prod}} := h_r^{\text{fifth}} \cdot h_{(r+1) \bmod 12}^{\text{fifth}} \quad (5.5)$$

for $r \in [0 : 11]$. At this stage, we are only interested in the root note and not in the mode of the final chord and, thus, ignore this chord's third note.¹ Since the majority of classical pieces' final chords—independently of their mode—contain strong energy in the root as well as in the fifth chroma, this procedure provides the final chord's root with a high reliability:

$$r^{\text{root}} := \arg \max_{r \in [0:11]} h_r^{\text{prod}} \quad (5.6)$$

For monophonic endings, this method also works well since the third partial of the root always produces some energy in the fifth chroma (compare Section 2.2). Figure 5.3 shows the root note estimation for a piano example.

To obtain likelihoods for each pitch class being the final root, we calculate a vector $\mathbf{P}^{\text{tonic}} := (P_0^{\text{tonic}}, \dots, P_{11}^{\text{tonic}})^T$ of confidence measures using the Euclidean norm:

$$P_r^{\text{tonic}} := \frac{h_r^{\text{prod}}}{\ell_2(\mathbf{h}^{\text{prod}})} \quad (5.7)$$

with $r \in [0 : 11]$.

¹In classical music, the final chord may not be representative for the overall mode of the piece. For instance, many minor pieces end in the associated major chord (Picardy third). Furthermore, certain symphony movements show a development from a minor key to the parallel major key.

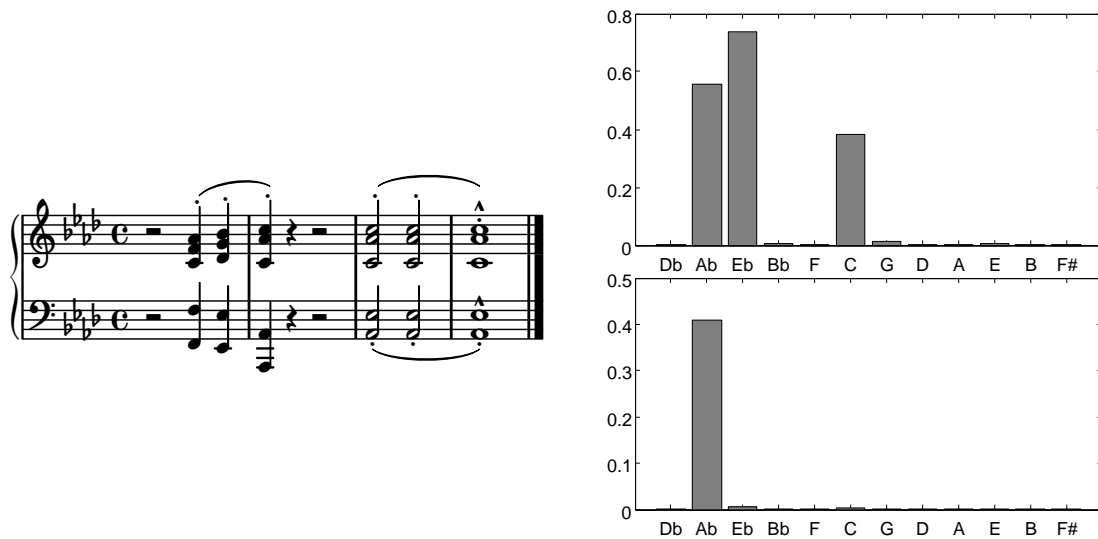


Figure 5.3. Final chord estimation process. The score denotes the last bars of F. Chopin’s Impromptu No. 1 for Piano, op. 29 in $A\flat$ major. The upper plot shows the re-sorted chroma histogram $\mathbf{h}^{\text{fifth}}$ from the last $F = 30$ frames. This results in the lower one \mathbf{h}^{prod} after pairwise multiplication. From this, we identify the correct root note $A\flat$ even though the maximum value in the chroma histogram belongs to $E\flat$.

5.1.2.3 Diatonic Scale Estimation

Since classical works or single movements may pass through certain tonal progressions, show parts in other keys, or even end in a key other than the global key,² we consider the full length of the recording to identify the underlying diatonic scale. Here, we assume that the most prominent diatonic scale corresponds to the global key’s major or natural minor scale, respectively. To this end, we extract a chroma histogram \mathbf{g}^{ℓ_2} from the whole piece and try to estimate the most probable diatonic scale. The concept of diatonic scales as “tonal levels” turned out useful for various tonal analysis tasks [69]. As an example, we denote G major as well as E minor as +1 level ($1\sharp$), $B\flat$ major and G minor as -2 level ($2\flat$). Since the diatonic scale consists of seven fifth-related notes (compare Figures 5.2 and 2.6), we again re-sort the histogram to a fifth ordering $\mathbf{g}^{\text{fifth}} := (g_0^{\text{fifth}}, \dots, g_{11}^{\text{fifth}})^T \in \mathbb{R}^{12}$. To obtain estimates for the different transpositions, we multiply each seven fifth-related chroma energies corresponding to the respective diatonic scale. We obtain the scale product histogram $\mathbf{g}^{\text{prod}} := (g_0^{\text{prod}}, \dots, g_{11}^{\text{prod}})^T \in \mathbb{R}^{12}$ by calculating

$$g_r^{\text{prod}} := \prod_{n=0}^{11} (g_n^{\text{fifth}})^{V_{(n-r+5) \bmod 12}}. \quad (5.8)$$

with $r \in [0 : 11]$. The template $\mathbf{V} := (V_0, \dots, V_{11}) \in \mathbb{R}^{12}$ is zero for the pitch classes outside the diatonic scale resulting in the multiplicative identity for these pitch classes. Later, we explain the details of this template (see Equation (5.10)).

Similar to Equation (5.7), we compute likelihood measures $\mathbf{P}^{\text{scale}} := (P_0^{\text{scale}}, \dots, P_{11}^{\text{scale}})^T \in \mathbb{R}^{12}$ for the diatonic scales:

$$P_r^{\text{scale}} := \frac{g_r^{\text{prod}}}{\ell_2(\mathbf{g}^{\text{prod}})} \quad (5.9)$$

²Most frequently, this is the parallel key (compare Section 2.7).

with $r \in [0 : 11]$. Hereby, P_r^{scale} indicates the likelihood for the scale $d := r - 5$. For example, P_1^{scale} denotes the likelihood for the level $d = -4$ (Ab major scale or natural F minor scale³).

To account for the individual relevance of the notes, we propose a weighting procedure⁴ by means of four different templates of exponents $\mathbf{V} := (V_0, \dots, V_{11})^T$:

$$\begin{aligned}\mathbf{V}^{(1)} &= (0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0)^T \\ \mathbf{V}^{(2)} &= (0, 0, 0, 0, 1, 3, 2, 1, 2, 3, 1, 0)^T \\ \mathbf{V}^{(3)} &= (0, 0, 0, 0, 3.75, 4.75, 3.00, 3.75, 4.25, 4.50, 3.75, 0)^T \\ \mathbf{V}^{(4)} &= (0, 0, 0, 0, 4.04, 5.87, 4.27, 3.51, 5.00, 4.57, 3.20, 0)^T\end{aligned}\tag{5.10}$$

In Equation (5.8), the seven entries V_4, \dots, V_{10} are the weighting exponents for the scale degrees corresponding to the solfège syllables

$$(V_4, \dots, V_{10}) \rightarrow (fa, do, sol, re, la, mi, ti),\tag{5.11}$$

independently from the transposition index r (compare also Figure 2.6 and Table 2.1). For example, V_4 is the weighting for the tonic note of the corresponding major scale.

With the exponents $\mathbf{V}^{(1)}$, we realize equal weighting. $\mathbf{V}^{(2)}$ emphasizes the notes of the tonic chords (for level 0, these are the CM and the Am chords). $\mathbf{V}^{(3)}$ results from Temperley's templates [235] by averaging the major and the relative minor profile for the diatonic scale notes. $\mathbf{V}^{(4)}$ is the same for Krumhansl's templates [124]. For all templates, we ignore the non-diatonic notes by exponentiating them with zero. Up to this strategy, the scale estimation step basically equals a common template matching approach.⁵ However, the multiplicative procedure leads to a suppression of those scales for that one or more scale notes have only small energy.

5.1.2.4 Decision Process

In order to select the most probable key, we combine every tonic note likelihood with the associated diatonic scale likelihoods:

$$\begin{aligned}P_r^{\text{major}} &= (P_r^{\text{tonic}})^s \cdot P_r^{\text{scale}} \\ P_r^{\text{minor}} &= (P_r^{\text{tonic}})^s \cdot P_{(r-3)}^{\text{scale}} \pmod{12}\end{aligned}\tag{5.12}$$

with $r \in [0 : 11]$. Here, the exponent $s \in \mathbb{R}$ serves as a tuning parameter between root and scale influence. To calculate the likelihood for C major, for example, we combine the likelihood P_5^{tonic} —for pitch class C being the root—with the likelihood P_5^{scale} for level $d = 0$ (no accidentals). For the minor case, we need to shift the scale vector by three entries to associate the roots with the corresponding minor scales. To compute the likelihood for C minor, we multiply P_5^{tonic} with the scale likelihood P_2^{scale} corresponding to the level $d = -3$.

³With this procedure, we do not consider harmonic or melodic minor scales, which may lead to a degraded scale estimation performance. We believe that, when analyzing whole movements, the notes of the natural minor scale are sufficiently present. Since we compute a global scale estimate, we therefore assume little effect of the alterations in minor keys. Mostly, this was confirmed by our experimental observations.

⁴Note that for a product calculation, we have to perform weighting by exponentiation instead of multiplication.

⁵Essentially, the fifth ordering is only for the purpose of convenient visualization. In this representation, all diatonic scale notes are neighbors.

Table 5.1. Contents of the dataset *Symph*. For each composer, we denote the numbers of the symphonies contained in the dataset.

<i>Composer</i>	<i>Symphonies No.</i>
Beethoven, L. van	2, 3, 8
Brahms, J.	2, 3
Bruckner, A.	3, 4, 8
Dvořák, A.	5, 7
Haydn, J.	22, 29, 60, 103
Mendelssohn-B., F.	3, 5
Mozart, W. A.	35, 39, 40, 41
Schubert, F.	2, 3, 8
Schumann, R.	2, 4
Sibelius, J.	3, 4
Tchaikovsky, P. I.	5, 6

We calculate a combined likelihood vector $\mathbf{P}^{\text{comb}} \in \mathbb{R}^{24}$ by concatenating major and minor estimates:

$$\mathbf{P}^{\text{comb}} = (P_0^{\text{comb}}, \dots, P_{23}^{\text{comb}})^T := (P_0^{\text{major}}, \dots, P_{11}^{\text{major}}, P_0^{\text{minor}}, \dots, P_{11}^{\text{minor}})^T \quad (5.13)$$

From this, we obtain the key by taking the index with the maximal likelihood:

$$k^{\text{key}} = \arg \max_{k \in [0:23]} P_k^{\text{comb}} \quad (5.14)$$

5.1.3 Evaluation

5.1.3.1 Datasets

For evaluating our algorithm, we consider three datasets of classical music audio recordings. The first one (*Symph*) contains symphonies from eleven classical and romantic composers (all movements for each symphony), 29 symphonies with 115 tracks in total. We compiled this data from commercial recordings. Table 5.1 lists the composers and works.

The second dataset (*SMD*) is a selection from Saarland Music Data Western Music, a freely available dataset collected in a collaboration of Saarland University and MPI Informatik Saarbrücken with Hochschule für Musik Saar [169]. This data encompasses music for solo instruments, voice and piano, as well as chamber and orchestral music. We annotated the key for the 126 tracks showing clear tonality.⁶

Third, we test our method on a dataset of piano music recordings (*Pno*). The authors of the publications [183] and [239] used this data to investigate key determination. This allows for a direct comparison of key detection performance. The set contains commercial audio recordings of 237 piano pieces by Bach, Brahms, Chopin and Shostakovich. The composers explicitly dedicated these pieces to a special key such as, for example, in “The Well-Tempered

⁶To this end, we skipped works of Bellini, Berg, Debussy, Donizetti, Martin, Poulenc and Ravel as well as the first and second movement of Faure’s op. 15. From Schumann’s works, we removed Op. 15 and Op. 48 since they are work cycles and do not constitute sets of separate pieces in some way. For detailed information, see <http://www.mpi-inf.mpg.de/resources/SMD>. Our key annotations are also available on this website.

Table 5.2. Properties of the key evaluation datasets. The first rows summarize the distribution of the modes. The middle part outlines the final chord statistics. Last, we show the overlap of final chord and global key labels throughout the datasets.

<i>Dataset</i>	<i>Symph</i>	<i>SMD</i>	<i>Pno</i>	<i>Total</i>
Major global key	70 %	57 %	49 %	56 %
Minor global key	30 %	43 %	51 %	44 %
Major final chord	72 %	55 %	70 %	67 %
Minor final chord	12 %	20 %	14 %	15 %
Third-less final chord	16 %	25 %	15 %	18 %
Final chord $\hat{=}$ global key	70 %	64 %	53 %	60 %
Final root $\hat{=}$ global tonic	99 %	98 %	98 %	99 %

Clavier”, which contains each one prelude and fugue for every key. Pauws [183] provides detailed information about the recordings.

Table 5.2 shows some properties of the datasets. Final chord and global key coincide for only 60 % of the pieces. However, the final chord’s root matches the global key’s tonic note almost always (99 %). Most of the mode deviations are Picardy thirds (20 %) with a minor piece ending in the relative major chord (the opposite case is rare—see Section 2.7). The remaining exceptions stem from third-less final chords (18 %) such as empty fifths (1 %) or unisono endings (17 %). Overall, 71 % of the pieces end in a full triad while 11 % end in a fifth-less chord.

5.1.3.2 Experimental Results

We investigate the influence of the system parameters in a detailed study on the three datasets *Symph*, *SMD*, and *Pno*.⁷ Table 5.3 shows an overview of these results. The last column denotes the average performance Λ_{Total} , computed as a weighted sum over the performance on the three individual datasets:

$$\Lambda_{\text{Total}} = \frac{115 \Lambda_{\text{Symph}} + 126 \Lambda_{\text{SMD}} + 237 \Lambda_{\text{Pno}}}{478} \quad (5.15)$$

First, we test different sizes F of the final frame set. Here, a value of $F = 20$ frames corresponding to 2 s duration performs best. This value seems to balance the requirements for short final chords (no failures caused by previous chords) with a sufficiently high robustness. To estimate the individual influence of root and scale estimation, we run the algorithm with different weight exponents s for the decision process. A slight preference of the scale confidence with $s = 0.8$ yields best results. Next, we show selected results for different energy threshold factors ρ . For this parameter, a value of $\rho = 0.20\%$ seems to optimally separate silence from music frames. With this low dynamic threshold, we may also include the reverberation of the final chord to a certain extent. For this reason—and, because of the frequent occurrence of a final ritardando in classical music performances—we do not have to

⁷The results presented in this section slightly deviate from the numbers in [252]. This is due to a misinterpretation in the paper where we assumed a false sampling rate for the chroma computation. Because of that, the optimal number for the parameter F is lower here. Furthermore, we test the effect of reducing the contribution of the bass region to the chroma features, which shows a similar effect as the sampling rate confusion.

Table 5.3. Correct full key classification results for different parameter sets. We test the influence of the size of the final frame set F (A), the root-scale weight exponent s (B), the energy threshold factor ρ (C), and the weight exponent set \mathbf{V} (D). The bold lines mark the best results for each parameter. For (E1), we removed the multiplication in the tonic note estimation Equation (5.5). For the results (E2), we replaced the product in Equation (5.8) with a weighted sum. (E3) considers both of these changes. For (E4), we used the averaged (major and relative minor) Krumhansl templates—without restriction to the diatonic entries—and calculate a sum instead of a product. (E5) and (E6) constitute the standard template matching procedure for 24 keys using the templates proposed by Krumhansl (E5) and Gómez (E6), respectively.

<i>Parameters</i>	<i>Symph</i>	<i>SMD</i>	<i>Pno</i>	<i>Total</i>
A)	$s = 0.8, \rho = 0.15\%, \mathbf{V} = \mathbf{V}^{(2)}$			
$F = 10$	90.4 %	95.2 %	92.8 %	92.2 %
$F = 20$	92.2 %	95.2 %	94.1 %	93.9 %
$F = 30$	92.2 %	92.1 %	93.2 %	92.7 %
$F = 40$	93.0 %	92.1 %	93.7 %	93.1 %
$F = 60$	92.2 %	89.7 %	92.0 %	91.4 %
B)	$F = 20, \rho = 0.15\%, \mathbf{V} = \mathbf{V}^{(2)}$			
$s = 0.6$	92.2 %	93.7 %	94.9 %	93.9 %
$s = 0.7$	92.2 %	94.4 %	94.5 %	93.9 %
$s = 0.8$	92.2 %	95.2 %	94.1 %	93.9 %
$s = 0.9$	92.2 %	94.4 %	94.1 %	93.7 %
$s = 1.0$	92.2 %	94.4 %	93.7 %	93.5 %
$s = 1.2$	92.2 %	94.4 %	93.7 %	93.5 %
C)	$F = 20, s = 0.8, \mathbf{V} = \mathbf{V}^{(2)}$			
$\rho = 0.10\%$	91.3 %	94.4 %	94.9 %	93.9 %
$\rho = 0.15\%$	92.2 %	95.2 %	94.1 %	93.9 %
$\rho = 0.20\%$	93.0 %	94.4 %	94.1 %	93.9 %
$\rho = 0.25\%$	93.9 %	92.9 %	94.1 %	93.7 %
$\rho = 0.30\%$	93.9 %	92.9 %	92.8 %	93.1 %
$\rho = 0.50\%$	93.9 %	92.1 %	92.8 %	92.9 %
D)	$F = 20, s = 0.8, \rho = 0.20\%$			
$\mathbf{V} = \mathbf{V}^{(1)}$	88.7 %	93.7 %	92.8 %	92.1 %
$\mathbf{V} = \mathbf{V}^{(2)}$	93.0 %	94.4 %	94.1 %	93.9 %
$\mathbf{V} = \mathbf{V}^{(3)}$	90.4 %	93.7 %	95.8 %	93.9 %
$\mathbf{V} = \mathbf{V}^{(4)}$	89.6 %	93.7 %	95.4 %	93.5 %
E)	$F = 20, s = 0.8, \rho = 0.20\%$			
<i>E1</i>	83.5 %	80.2 %	81.0 %	81.4 %
<i>E2</i>	88.7 %	92.1 %	89.0 %	89.7 %
<i>E3</i>	74.8 %	63.5 %	57.4 %	63.2 %
<i>E4</i>	88.7 %	90.5 %	86.5 %	88.1 %
<i>E5</i>	44.3 %	42.1 %	49.4 %	46.2 %
<i>E6</i>	71.3 %	75.4 %	65.0 %	69.2 %

worry about choosing a fixed small number of final frames F independently of the tempo of the piece.⁸

⁸Intuitively, one might consider longer frame sets for slow pieces and shorter ones for fast pieces. However, we did not find any increase of performance using such method.

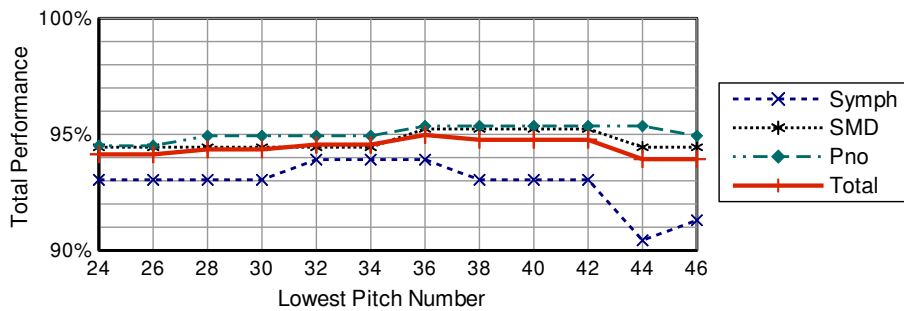


Figure 5.4. Key detection results for different pitch ranges. In this study, we vary the lower pitch boundary for computing the chroma features. We obtained best performance with a lower limit at $p = 36$.

Testing the weight exponents \mathbf{V} , the emphasis of the chord notes in $\mathbf{V}^{(2)}$ and the template derived from Temperley $\mathbf{V}^{(3)}$ perform best. To check the influence of the individual steps, we perform several experiments without the multiplicative procedure in the tonic note estimation or in the diatonic scale estimation (block (E) in Table 5.3). For the first test ($E1$), we remove the pairwise fifth-multiplication for the tonic note estimation and directly pick the maximum from the final chroma frames, leading to a decrease of about 10 percentage points in performance. In contrast, replacing the product matching for the scale (Equation (5.8)) with a weighted sum only slightly affects performance. The combination of both changes ($E3$) leads to considerably decreasing results. For ($E4$), we extended this weighted sum to all twelve pitch classes by combining the values of Krumhansl’s templates for the major and relative minor keys. This still leads to good results of 88% total performance. However, the traditional template matching ($E5$) with the 24 Krumhansl probe tone ratings—without considering the final tonic note—performed much worse (46% on average). Inspired by Schaab [211], we test this approach using Gómez’ templates instead [76]. This results in a performance of about 70%—much better than with Krumhansl’s templates but still worse than our final chord algorithm. Overall, the use of the final chord approach turns out beneficial for key detection performance compared to pure template-based strategies. The fifth multiplication in the tonic note estimation step seems to be important whereas the details in the diatonic scale matching procedure shows less influence.

Most of the parameters that we evaluated here show important impact especially on one of the databases. A reason for this may be the different acoustic behavior of orchestra and piano recordings. Furthermore, different properties of the music contained in the datasets may play a role. For example, the style dependency of tonality (compare Table 5.5) or the temporal dimensions in symphonic music in contrast to soloistic and chamber music may affect the difficulty of key detection.

Inspired by some observations in the testing procedure, we conducted a further study to estimate the lower pitch boundary for the chroma computation (Equation (3.22)). In the Chroma Toolbox algorithm [165], the default boundary is the pitch corresponding to the lowest piano key ($p = 21$). Instead of using this value, we vary the lower pitch boundary from $p = 24$ up to $p = 46$. The individual datasets react on the variation of this parameter in different ways. For *Symph*, a lower value of about $p = 34$ leads to best results whereas for the other datasets, higher boundaries of roughly $p = 40$ seem to be optimal. This behavior may arise due to the acoustic properties of the low piano keys (*Symph* does not contain piano music). Regarding the total performance Λ_{Total} , we found $p = 36$ to be an optimal boundary. For the upper boundary, we keep the default value $p = 108$ since we found no considerable effect by changing this value.

Table 5.4. Key extraction results for the optimal parameter combination. As evaluated in the previous experiments, we choose the optimal parameters $F = 20$, $s = 0.8$, $\rho = 0.20\%$, $\mathbf{V} = \mathbf{V}^{(2)}$ and a low pitch boundary at $p = 36$.

<i>Dataset</i>	<i>Symph</i>	<i>SMD</i>	<i>Pno</i>	<i>Total</i>
Correct full key	93.9 %	95.2 %	95.4 %	95.0 %
Correct tonic note	100 %	96.8 %	96.6 %	97.5 %
Fifth errors	0 %	0.8 %	1.3 %	0.8 %
Third errors	0 %	2.4 %	2.1 %	1.7 %
Mean confidence	96.5 %	96.1 %	97.1 %	96.7 %

Table 5.5. Results by historical period. The parameters are the same as for Table 5.4.

<i>Period</i>	Baroque	Classical	Early Rom.	Late R./Mod.
No. in <i>Symph</i>	0	46	26	43
No. in <i>SMD</i>	11	49	20	46
No. in <i>Pno</i>	144	0	0	93
Total No.	155	95	46	185
Correct full key	97 %	96 %	96 %	93 %
Correct tonic note	97 %	98 %	100 %	97 %

In Table 5.4, we show individual error rates for the best parameter set. Hereby, we emphasize the small number of fifths errors that arise frequently for other key detection approaches. Third errors include all tonic note relations of minor and major thirds, including the relative key. Especially on symphonic data, identification of the correct tonic note is clearly more reliable than full key detection.

In Table 5.5, we break down these results to the historical periods. To this end, we cluster the results by composer and aggregate music by Bach (Baroque), Haydn, Mozart and Beethoven (Classical), Schubert, Schumann and Mendelssohn (Early Romantic), and the rest (Late Romantic and Modern). As expected, we find lower accuracy for the late romantic and modern pieces. This may proceed from a higher tonal complexity in these periods.

The results for the optimal parameter combination (Table 5.4) are slightly below the state-of-the-art [39, 174]. Hereby, we have to take into account that the authors of these papers evaluated their algorithms on music from one composer for one type of orchestration. Our data comprises a wider range of styles and instrumentations. On the *Pno* set, we almost reach the accuracy of 98 % presented in [239]. To compare to a public algorithm, we run the key detection algorithm of MIRtoolbox from University of Jyväskylä [128] on our data with a total performance of 67.5 %. This method is a common template matching approach based on chroma features. Here, the authors use Gómez’ templates for the key estimation. Looking at the results in Table 5.6, we see that our method performs better for full key detection. Especially, the final-chord-based algorithms outperforms the template-based approach with respect to the tonic note estimation performance. In our re-implementation of the template matching, we obtain roughly similar results (69.2 %) when using the same templates (setting (*E6*) in Table 5.3). The deviations between the results of the MIRtoolbox and our template matching may originate from a different chroma extraction method.

To further compare the performance of the proposed algorithms with other methods, we performed another study [259] by re-implementing several published algorithms. Beyond the

Table 5.6. Results of the MIRtoolbox key detection algorithm. For this experiment, we use the public algorithm presented in [128].

<i>Dataset</i>	<i>Symph</i>	<i>SMD</i>	<i>Pno</i>	<i>Total</i>
Correct full key	73.0 %	71.2 %	62.9 %	67.5 %
Correct tonic note	78.4 %	71.2 %	62.9 %	68.8 %
Fifth errors	9.0 %	12.8 %	13.1 %	12.0 %
Third errors	12.6 %	14.4 %	20.2 %	16.8 %

standard template matching approach, we consider the idea by Van de Par *et al.* [239]. They used a profile learning strategy together with a special weighting of the beginning and ending phases. To account for approaches using geometrical pitch models, we also test the symmetry model by Gatzsche and Mehnert [72]. In [157], they evaluated this model for key detection.

For estimating the optimal parameters, we run each algorithm with different parameter settings in a stepwise fashion. To that end, we optimize each parameter by maximizing the weighted total performance Λ_{Total} and fix the remaining parameters to default or best fit values. We perform this overfitting on the three datasets *Symph*, *SMD*, and *Pno* since we later use an unseen dataset (*Cross-Era*) for evaluation.

For the basic chroma features, we test six different implementations (compare Section 3.5.3): CP, CLP (with $\eta = 1000$), CRP, HPCP, EPCP (three iterations of the harmonic product spectrum), and NNLS. We obtain the following results for the different algorithms (for the parameters' meaning, see [259]):

- **Template matching.** We test the profiles proposed by Krumhansl [124], Temperley [235], and Gomez [76] with the latter ones performing best. Although Gómez developed these profiles in combination with HPCP features, NNLS features outperform these features (84.7%), followed by CLP.
- **Profile learning.** For the profile training, we perform a cross-validation with 98% training data, 2% test data, and 5000 repetitions, exactly following [239]. We find best performance for CLP chroma features (92.3%)—closely followed by NNLS. We cannot reach the result presented in [239] (98% on the *Pno* dataset). As a reason for this, we assume that the specific chroma implementation presented in that work (including a masking model) provides additional benefits.
- **Symmetry model.** This algorithm [157] works best in conjunction with NNLS chroma. We find the optimal pitch set energy threshold at $f_{\text{TR}} = 0.12$. The angular vector value comes out best at $w_{\text{sym}} = 0.53$ leading to a total performance of 82.6%.
- **Final chord.** For the final chord algorithm, we found a slightly deviating optimal parameter set here:

$$F = 19, \quad s = 0.9, \quad \rho = 0.19\%, \quad \mathbf{V} = \mathbf{V}^{(2)} \quad (5.16)$$

With these parameters, we obtained 93.7% accuracy. The final chord algorithm obtained optimal results on the basis of CP chroma features. Here, we do not test the influence of the lower pitch boundary in the chroma computation step but use the piano range $p \in [21 : 108]$.

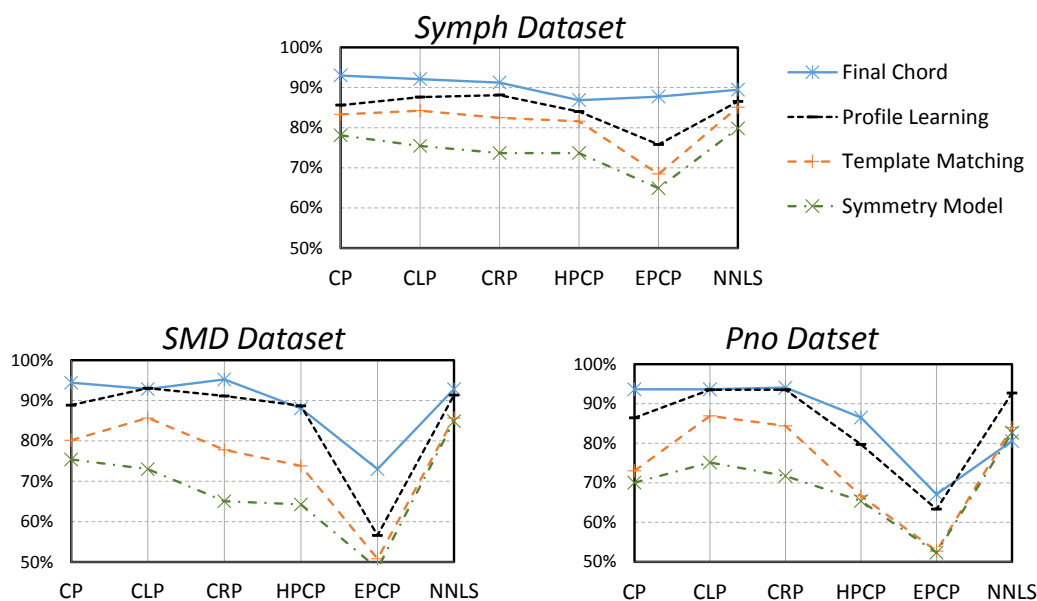


Figure 5.5. Evaluation of different key detection algorithms. Here, we show the individual key recognition accuracies for the three datasets *Symph*, *SMD*, and *Pno*. We compare six types of chroma features serving as basis feature for the different methods.

In Figure 5.5, we show the overall results of the key detection evaluation for different types of chroma features. All algorithms considerably depend on the chroma extraction method—especially when the data includes piano music (*Pno* and *SMD*). NNLS features often obtain best results and seem to be the most stable basis for key detection methods. EPCP features are not a good choice for this purpose. The profile learning and the final chord strategies perform similarly. Hereby, the first one is rather data-dependent whereas the final chord algorithm requires a fine parameter tuning.

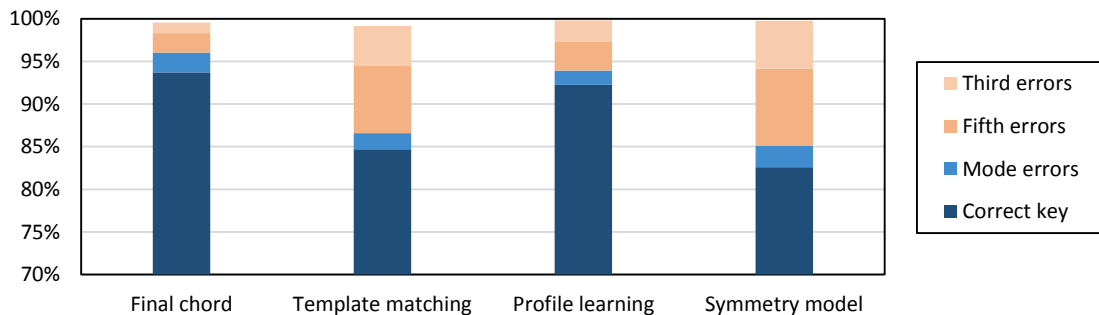
In the previous experiments, we optimized the parameters of the algorithms with respect to the evaluation datasets. To estimate the real-world performance of these algorithms, we make use of unseen data. To this end, we use a subset of the *Cross-Era* dataset (see Section 7.1). We annotated the key for 1200 pieces comprising both piano and orchestral music from the periods Baroque, Classical, and Romantic. For each method, we use the feature and parameter setting performing best in the previous experiments.⁹ We obtain a performance of **83.9%** for the template matching algorithm, **87.1%** for the profile learning, **80.4%** for the symmetry model, and **85.4%** for the final chord approach. Figure 5.6 displays the detailed results. Compared to the optimization datasets, the overall performance is worse and the differences between the methods are smaller. Profile learning and final chord still obtain the best results. However, the learning strategy seems to be slightly more robust than the parameter-dependent final chord algorithm.

5.1.4 Conclusion

In this section, we presented a new rule-based approach to extract the global key information from classical music audio recordings. The method puts special emphasis on the final chord of the piece. After extracting chroma features, we automatically select a set of final frames

⁹For the profile learning approach, we also train the profiles on the previously used datasets *Symph*, *SMD*, and *Pno*.

a) *Symph*, *SMD*, *Pno* datasets (optimized parameters)



b) *Cross-Era* dataset

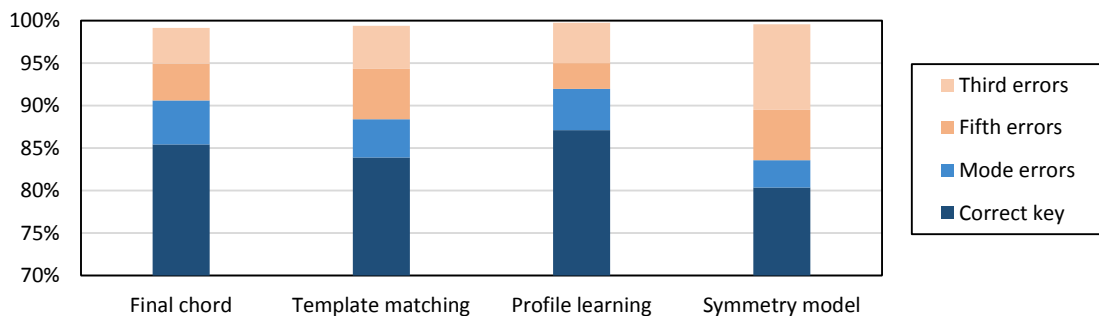


Figure 5.6. Key detection performance for unseen data. Here, we compare the key detection results on the datasets previously used with the performance on unseen data. The upper plot shows the weighted total performance on the three datasets *Symph*, *SMD*, and *Pno*. In the lower plot, we display the results on a 1200 track database of orchestra and piano music (*Cross-Era* database).

that exceed a given energy threshold. From these frames, we determine the final chord’s root using a pairwise multiplication of fifth-related chroma values. From chroma statistics of the full piece, we estimate the underlying diatonic scale. Finally, we combine these results by multiplying corresponding likelihood measures obtaining the full key.

For the evaluation, we consider three datasets on symphonic, chamber, and solo piano music containing 478 recordings in total. We performed a detailed study to estimate the optimal parameters for our algorithm. In these experiments, we reach success rates of up to 95 % for full key detection and 97 % for tonic note detection. Our results are in the range of most state-of-the-art approaches designed for key detection in classical music. To compare with these public methods, we re-implemented several key detection systems proposed in the literature. For all algorithms, we found a considerable dependency on the chroma feature type. Hereby, CP and NNLS features performed best. On unseen data, we tested the robustness of the methods. Compared to the most competitive approach by Van de Par *et al.* [239], the final-chord-based algorithm seems to be slightly less robust.

5.2 Local Estimation of Scales

5.2.1 Introduction

As we saw in Section 2.7 and in the previous section, we do not find a constant global key for every musical piece. Rather, composers use to play with key and key expectation and create transitional phases (modulations) leading from one local key to another. For classical

music, these transitions usually take place over a considerable span of time. Hence, it is often hard to manually annotate a ground truth segmentation of local keys since the segment borders—and even the keys—are often ambiguous. For these reasons, we restrict ourselves to a visualization of modulations in this section without performing any quantitative evaluation. Our approach relates to music theory concepts on harmony and tonality. In particular, we consider scale-based theories for explaining tonal relations (compare Sections 2.5 and 2.7) and derive automatic analysis methods based on these theories.

The first visualization type presented in this section serves to display the temporal evolution of local keys within a movement (Section 5.2.4). This method relates to Gárdonyi’s and Nordhoff’s [69] analysis technique regarding diatonic key relationships and “tonal levels.” We calculate local estimates for the underlying diatonic scales and arrange these scale estimates according to a perfect fifth series in order to account for tonal similarity of pitch classes. Visualizing the local results over time provides a useful overview of the modulation structure of a piece.

In Section 5.2.5, we present a second method referring to the general scale type and the symmetries of the local pitch content. This technique relates to scale-based theories of harmony such as the distance principle by Gárdonyi, Nordhoff, and Lendvai [69] or the Tonfeld concept by Simon [82]. Scale models such as the whole tone scale, the octatonic scale, or the acoustic scale play an important role in impressionistic music or in O. Messiaen’s compositions, among others. With our method, we compute the local likelihood for different scale types. We display these estimates over the course of a piece in order to show the locally prominent scales. This allows for an analysis of the formal aspects of tonality.

Both visualization techniques may be helpful for assisting musicological research. With such an automatic approach, it is possible to get a quick overview of a piece with respect to tonal relationships and progressions. This also applies to particularly long works such as operas or symphonies where the analysis of large-scale structures may be very costly. Furthermore, an automatic approach enables the search after tonal phenomena on large musical corpora and their statistical analysis.

This section closely follows the study presented in [257]. From a musicological point of view, Habryka [83] published a case study using some of these methods for analyzing a particular piece of late romantic music (the Scherzo from H. Rott’s first symphony). Beyond that, we published a key segmentation method for pop music based on a very similar method [253], which we do not consider here.

5.2.2 Musicological Foundations

The analysis technique presented in this work relies on the local scale material used in a composition. In Western music theory from the 19th century on, there are two ways of treating scales and their relation to tonality. Some scholars consider chords and chord progressions as fundamental—without focussing on the pitch class content [42,51,90,200,214]. Understanding harmony this way, a scale is the consequence of the used chords. Other musicologists consider scales as preexistent and deduce the chords as triads on the scale degrees [69,82,133,219,249]. In Section 2.7, we already discussed these contrary notions.

Besides such local observations, our visualization method allows for analyzing the formal aspects of tonality. In Schenkerian analysis [212], a piece of music constitutes a sequence of scale degrees (“Stufen”). Hereby, we understand the term “scale degree” in an extended and more abstract way. It no longer denotes a single note or triad but consumes several harmonies that constitute autonomous chords themselves. These scale degrees are prolonged and

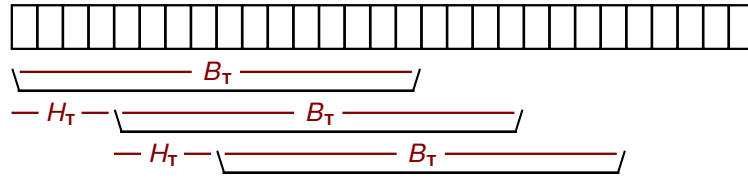


Figure 5.7. Segmentation of a chromagram. Each box stands for one chromagram frame. We divide the initial chromagram into analysis windows with a blocksize B_T and a hopsize H_T given in frames.

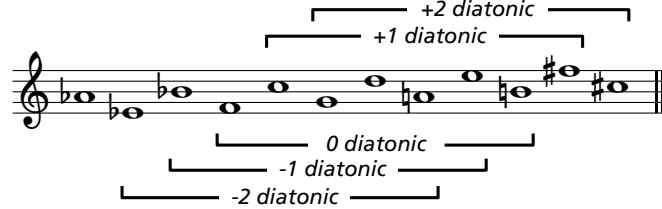


Figure 5.8. Diatonic subsets of a chromatic scale. We notate the chromatic scale in a perfect fifth ordering. The brackets are marking several diatonic subsequences. We name the scales according to the absolute fifth measurement. Diatonic scales with a close relation share a high number of pitch classes. Compare Section 2.5 for a more profound discussion.

connected to formal concepts such as the sonata form or the fugue. Other theories emphasize the structural purpose of tonality [22]. Further large scale analyses of tonality focus on the music dramas of R. Wagner—such as the analyses performed by Lorenz [143], which relate to our visualization method of local keys. The idea of aggregating pitch classes to superior tonal structures influenced recent musicological concepts such as the pitch class set theory [64].

5.2.3 Feature Extraction

Similar to the previous section, we build our local tonality visualization method on chroma feature representations of the audio data Section 3.5. Here, we use a normalized chromagram \mathcal{C}^{ℓ_1} based on the CP chroma implementation from the public Chroma Toolbox [165]. After applying a tuning estimation step, we compute a log-frequency spectrogram $\mathcal{Y}(p, m)$ in the piano range $p \in [21 : 108]$ with frame index $m \in [0 : M - 1]$. From this, we derive the chromagram $\mathcal{C}^{\ell_1}(q, m)$ with $q \in [0 : 11]$ as described in Equation (3.22).

We compute the chroma vectors with an initial feature rate of $f_{\text{feat}} = 10$ Hz. For analyzing the local pitch content, we need larger analysis windows. Therefore, we group the chroma vectors to blocks of size B_T with a hopsize of H_T such as shown in Figure 5.7. A block of $B_T = 200$ feature frames corresponds to an analysis window of $B_T/f_{\text{feat}} = 20$ s. For every block containing B_T chroma vectors, we compute a chroma histogram \mathbf{g}^{ℓ_1} as presented in Section 3.5.5.

To account for harmonic similarity of pitch classes, it turned out useful to re-order the chroma vector to a series of perfect fifths (D \flat , A \flat , E \flat , \dots , F \sharp). For each block, we obtain a fifth-ordered chroma histogram $\mathbf{g}^{\text{fifth}} := (g_0^{\text{fifth}}, \dots, g_{11}^{\text{fifth}})^T \in \mathbb{R}^{12}$ as introduced in Section 5.1.2.

5.2.4 Analysis of Modulations

The first analysis method proposed in this section refers to the local key of the music. For this, we consider the analysis method presented in [69] regarding the similarity of fifth-related

keys. By re-ordering the chromatic scale to a series of perfect fifth related pitch classes, a diatonic scale corresponds to an excerpt of seven neighbors (for convenience of the reader, we repeat in Figure 5.8 the illustration from Section 2.5). In such a representation, two fifth-related diatonic scales such as the C major and the G major scale only differ by one note (in this example, F \sharp instead of F). We use the nomenclature presented in [69] and denote the diatonic scales according to the number and type of accidentals necessary for notation. For example, a D major scale (2 \sharp) is called +2 diatonic or +2 level, an Ab major scale is a -4 diatonic. Beyond this **absolute fifth measurement**, which denotes the scales in accordance with the required accidentals, it is sometimes more convenient to use **relative fifth measurement**. Here, level 0 indicates the diatonic scale corresponding to the global key. The other scales obtain their names from the relative distance \mathcal{D} to the global key. In Section 2.5, we discussed the musical properties of diatonic scales in more detail.

Similar to Section 5.1.2.3, we try to estimate the underlying diatonic scale. To do this for the *local* tonal content, we compute the local chroma histogram $\mathbf{g}^{\text{fifth}}$ for each analysis block. From this, we multiply each seven entries h_r corresponding to the seven pitches of a diatonic scale. The absence of one or more scale notes results in a multiplication with a small number and, thus, leads to a small likelihood for this scale. Following Equation (5.8), we calculate the estimates g_r^{prod} via

$$g_r^{\text{prod}} := \prod_{n=0}^{11} \left(g_n^{\text{fifth}} \right)^{V_{(n-r+5) \bmod 12}} \quad (5.17)$$

with $r \in [0 : 11]$. Hereby, g_r^{prod} describes the likelihood for the (absolute) level $d := r - 5$.

Inspired by the experimental results of Section 5.1, we weight the scale degrees with a set of exponents \mathbf{V} to account for the individual importance of the scale notes. This exponential weighting turned out to improve scale estimation in the context of global key detection (Section 5.1). We derive the specific template $\mathbf{V}^{(5)}$ from the Krumhansl tone profiles $\mathbf{V}^{(4)}$ [124] combined with a weighting of the tonic triads $\mathbf{V}^{(1)}$ (see Equation (5.10)):

$$\mathbf{V}^{(5)} = \frac{1}{2} \left(\mathbf{V}^{(1)} + \frac{1}{2} \cdot \mathbf{V}^{(4)} \right) = (0, 0, 0, 0, 1.51, 2.97, 2.07, 1.38, 2.25, 2.64, 1.30, 0)^T \quad (5.18)$$

We do not consider the off-scale notes and, thus, exponentiate them with zero. The proposed procedure corresponds to a multiplicative version of common template matching strategies. This turned out useful for obtaining a robust scale estimation algorithm.

Finally, we normalize \mathbf{g}^{prod} with respect to the ℓ_2 norm in order to obtain the diatonic scale likelihoods:

$$P_{\text{diatonic}}(d) = \frac{g_{d+5}^{\text{prod}}}{\ell_2(\mathbf{g}^{\text{prod}})} \quad (5.19)$$

with the level index $d \in [-5 : 6]$ indicating to the number and type of accidentals. With the normalization, we force the system to decide on the likeliest local diatonic scale (or combination of scales) even if all g_r^{prod} are rather small. This turned out to enhance the robustness of the method. As a drawback, the output for non-diatonic music is not always meaningful and, thus, we have to carefully consider the preconditions for applying this analysis method. For example, the presence of melodic or harmonic minor scales may produce misleading results. In the following, we will discuss such problems.

We now want to show a number of different analyses and discuss the characteristics of our method on the basis of several visualizations. Since diatonic scale estimation mainly relates

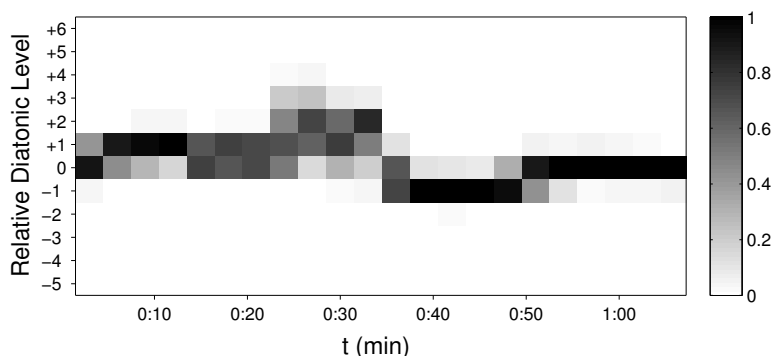
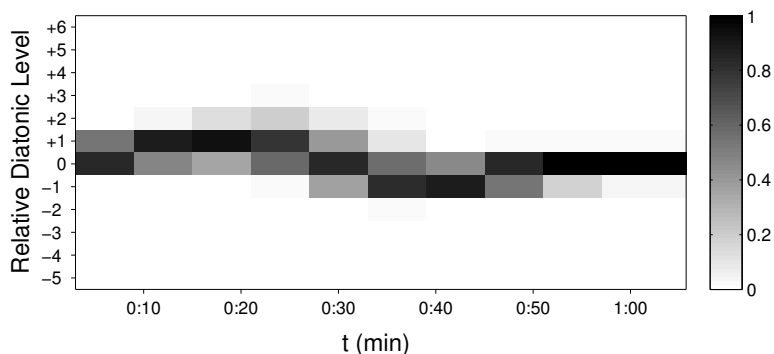
a) $B_T = 120, H_T = 30$ b) $B_T = 240, H_T = 60$ 

Figure 5.9. Diatonic scale visualization of J. S. Bach’s *Sinfonia No. 3, BWV 789*. For this piece in D major, the diatonic level 0 corresponds to $2\sharp$. We compare two different time resolutions: blocksize $B_T = 120$ frames and hopsize $H_T = 30$ frames in the upper plot (a), $B_T = 240$ frames, $H_T = 60$ frames in the lower plot (b). We analyze a recording by J. Sebestyén (Naxos 1993).

to Gárdonyi’s theory, we first look at J. S. Bach’s *Sinfonia* in D major BWV 789, which is discussed in [69, p. 250]. Note that for such tonality analyses, the nomenclature of the diatonic scales refers to the global key (relative fifth measurement). For this example, we denote the diatonic scale corresponding to D major ($2\sharp$) as level 0, the A major scale ($3\sharp$) as +1 level, etc. In contrast to Gárdonyi’s approach, our automatic method cannot discriminate between major and relative minor keys.

Figure 5.9 shows the results of this analysis. Using a fine time resolution (upper plot), we observe the general modulation structure with local keys at +1 in the beginning and -1 in the second half. At about 0:30 min, we see sudden jumps to the +2 level, in contrast to [69]. Here, a short modulation to the key $F\sharp$ minor is taking place (cadence in Measure 14) introducing the pitches $G\sharp$ and $D\sharp$ (as part of the $F\sharp$ melodic minor scale). Using larger analysis windows (lower plot), these local alterations show less influence—leading to a sine-shaped structure similar to [69]. From this observations, we see that the analysis results are meaningful in general. Problems may arise from short-time local modulations as well as for non-diatonic scales such as the melodic minor scale. With our method, we do not account for the possible alterations in minor scales. This may lead to a misestimation with scales having more sharp accidentals. We can see this effect in Figure 5.9 a) at about 0:30 min. Hereby, the temporal resolution of the analysis windows plays a crucial role. With a coarser resolution such as in Figure 5.9 b), the algorithm does not produce this error. Here, the local chroma histograms seem to have sufficient influence of the natural minor scale’s notes. Nevertheless, a more flexible approach for dealing with minor scales should be considered for future work.

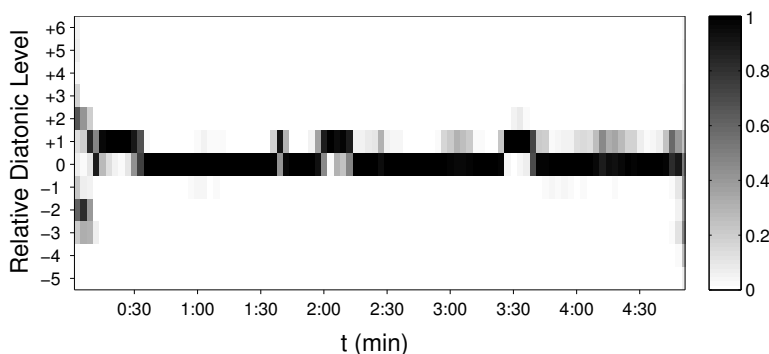


Figure 5.10. Diatonic scale visualization of G. P. da Palestrina’s “Missa Papae Marcelli.” We analyze the Kyrie from this mass with level $0 \hat{=} \text{no accidentals}$, $B_T = 100$, $H_T = 50$ in a recording by The Tallis Scholars (Gimell 1980/2005).

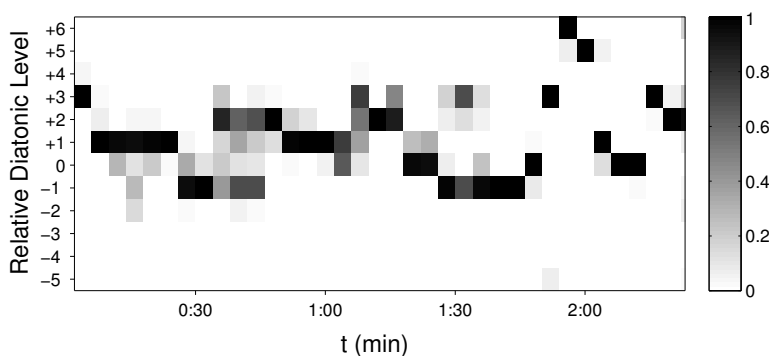


Figure 5.11. Diatonic scale visualization of O. di Lasso’s “Prophetiae Sibyllarum.” This plot shows No. 4 “Sibylla Cimmerica” from this work cycle, recorded by Ensemble Daedalus (Alpha 2005). Here, level $0 \hat{=} 1\flat$ according to the common notation, the final chord is GM. $B_T = 80$, $H_T = 40$.

Next, we want to discuss visualizations of pieces composed in various musical styles. In Figure 5.10, we show an analysis of the Kyrie from G. P. da Palestrina’s “Missa Papae Marcelli.” To a great extent, the pitch classes used in this piece belong to one diatonic scale. Smaller deviations to the +1 level arise due to local voice leading phenomena, for example, at 2:00 min where an $F\sharp$ is present. In contrast, the +1 scale detected at 3:30 min constitutes an ambiguity. Here, at the end of the “Christe eleison,” a GM triad holds for a couple of seconds. The algorithm misinterprets this half-cadence as a modulation to the +1 level. Further obscurities occur at the very beginning. After the initial silence, the voices come in gradually and, thus, the full scale material is present after some seconds for the first time. Due to this reason, scale detection is difficult here.

As a contrasting example, we display the analysis of a piece by the 16th century composer O. di Lasso (Figure 5.11). Here, the preconditions of scale-based diatonic music are not fulfilled. Sometimes, we find a small number of chords belonging to one diatonic scale. However, most of the chord changes rely on chromatic movements of the voices such as the change from an FM to an AM chord at 0:22 min. In such situations, the algorithm cannot estimate a constant scale since the chords are stemming from different diatonic scales. At about 2:00 min, we find an extreme example for this behavior. Overall, this example points to the limitations of our method for chromatic chord-based music.

In Figure 5.12, we show the analysis of a choral by J. S. Bach. We can recognize well the modulation to the +1 level in the repeated first phrase. The deviation to the “minus” region

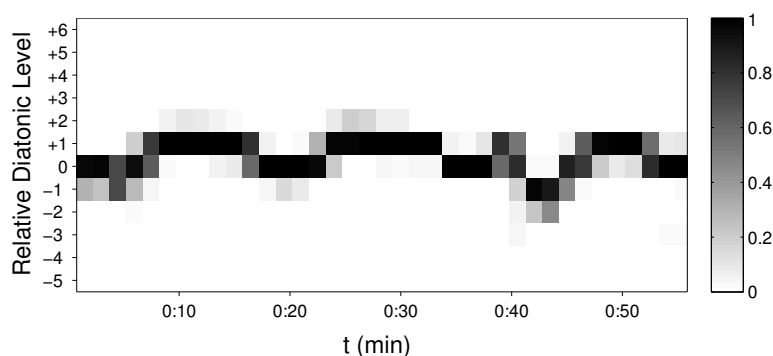


Figure 5.12. Diatonic scale visualization of a Choral from J. S. Bach’s “Johannespassion” BWV 245. We analyze the Choral No. 22 “Durch dein Gefängnis” in E major with level $0 \hat{=} 4\sharp$, $B_T = 42$, $H_T = 15$, in a recording by Scholars Baroque Ensemble (Naxos 1994).

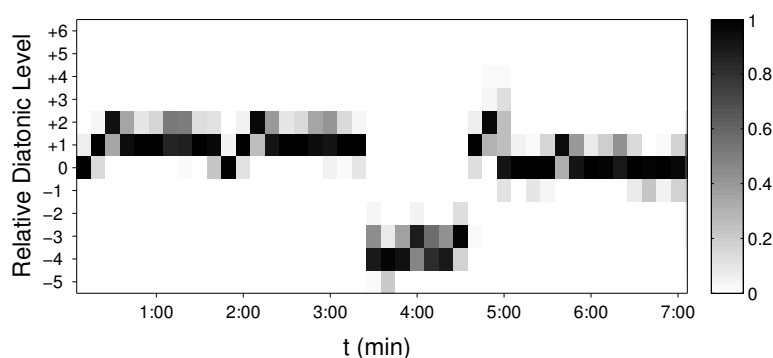


Figure 5.13. Diatonic scale visualization of a sonata by L. van Beethoven. This analysis describes the sonata Op. 14, No. 2, 1st movement in G major. Level $0 \hat{=} 1\sharp$, $B_T = 150$, $H_T = 60$, played by D. Barenboim (EMI 1998).

at about 0:40 min may arise due to the flat alterations at the chromatic elaboration of the text passage “unsere Knechtschaft.” The +1 level at 0:50 min is a misinterpretation of the long dominant triad BM.

Looking at Beethoven’s sonata Op. 14, No. 2 in G major (Figure 5.13), we observe the modulation shape of the classical sonata form with some interesting details. In the transition phase between the first to the second theme at 0:20 min (repeated at 2:00 min), we even see a small +2 area where we only expect level +1. Indeed, the piece modulates to A major for a short time, indicated by the presence of the pitch class $G\sharp$. In the development (3:30–5:00 min), we find keys in the minus region, in particular.

As the last example, we discuss R. Wagner’s overture from the opera “Die Meistersinger von Nürnberg” (Figure 5.14). Interestingly, we find a structure that roughly corresponds to the tonal shape of a sonata form. There are +1 regions in the first part, a highly modulating middle part, as well as an ending mainly based on level 0. The modulation path at about 4:00 min is remarkable, in particular. Here, our analysis indicates a modulation around the circle of fifths. After a short period at the levels +4 and +3, the tonal structure slowly leads back to the global key emphasized by a three minute coda mostly in level 0. For this particular example, the proposed method seems to provide an appropriate analysis. This has to be tested for other works by R. Wagner. Regarding larger structures such as R. Wagner’s tetralogy “Der Ring des Nibelungen,” a comparison of our algorithm’s output to the analyses

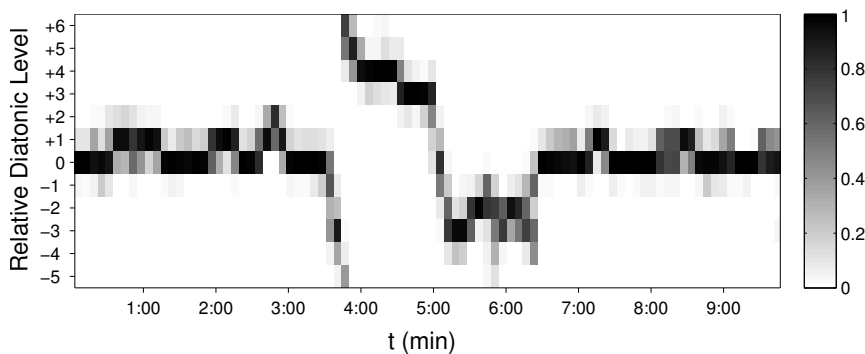


Figure 5.14. Diatonic scale visualization of R. Wagner’s “Meistersinger von Nürnberg.” For the Overture in C major, we show the progression of diatonic scales over time. Here, level 0 corresponds to no accidentals, $B_T = 150$, $H_T = 65$. The recording is played by the Polish National Radio Symphony Orchestra, conducted by J. Wildner (Naxos 1993).

presented in [143] could be of interest. A preliminary study on this subject can be found in [255].

5.2.5 Local Scale Type Estimation

As we discussed in Section 2.5, scale models other than diatonic scales play a crucial role in compositions from the late romantic period and the 20th century. To analyze which general scale types are present throughout a piece of music, we propose a second analysis method. Here, we do not compare the likelihood for different transpositions of one scale type. Instead, we only consider the likeliest transposition for every scale type and compare these maximal likelihoods among different scale types. To calculate the scale type estimates S_q , we again depart from the local chroma histogram \mathbf{g}^{ℓ_1} . Unlike Equation (5.17), we here use the chroma histograms in chromatic order. In the following, we use the abbreviation $\mathbf{g} = (g_0, \dots, g_{11})^T := \mathbf{g}^{\ell_1}$. We replace the exponents \mathbf{V} with binary templates $\mathbf{T} := (T_0, \dots, T_{11})^T \in \mathbb{R}^{12}$ describing the different scale models:¹⁰

$$S_q = \prod_{n=0}^{11} (g_n)^{T_{(n+q) \bmod 12}} \quad (5.20)$$

The index $q \in [0 : 11]$ indicates the transposition of the scale in semitones. We use the maximal value S_{\max} of all transpositions as scale type estimate:

$$S_{\max} = \max_q S_q \quad (5.21)$$

To investigate various concepts from music theory, we use templates \mathbf{T} for different scale models. We showed templates for several scales in Equations (2.24) and (2.25) such as the fifth-based pentatonic scale

$$\mathbf{T}^{\text{Pentatonic}} = (1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0)^T \quad (5.22)$$

or the symmetrical whole tone scale

$$\mathbf{T}^{\text{Wholetone}} = (1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0)^T. \quad (5.23)$$

¹⁰Note that the entries of the template vectors \mathbf{T} now refer to a chromatic pitch ordering.

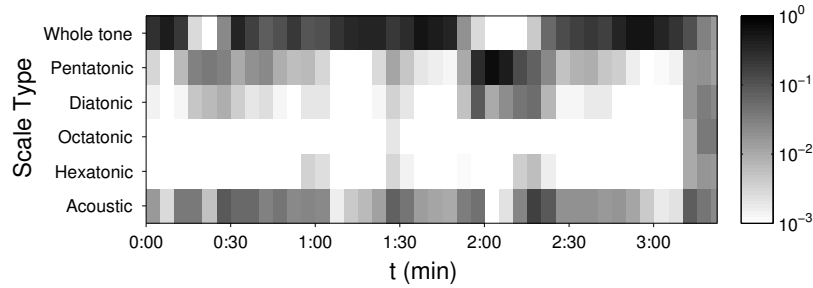


Figure 5.15. Scale type visualization of C. Debussy’s “Voiles.” For this piece (No. 2 from the first book of “Préludes” for piano), we estimate the presence of different scale types. The analysis windows exhibit a blocksize of $B_T = 100$ frames and a hopsize $H_T = 50$ frames. We consider a recording by F. Thiollier (Naxos 1998).

In Figure 2.10, we show these scales in Western music notation. For the symmetrical scales (whole tone scale, hexatonic scale, and octatonic scale), some of the transposed versions are identical to each other. Since we pick the maximum likelihood of all transpositions, this does not constitute a problem. However, in order to compare the likelihoods for different scale types to each other, we have to account for the varying number of notes $K_{sc} \in \mathbb{N}$ in the scales:

$$K_{sc} := \sum_{q=0}^{11} T_q. \quad (5.24)$$

We therefore introduce a normalization factor depending on the number of notes in the scale. For a ℓ_1 -normalized histogram $\mathbf{g} = \mathbf{g}^{\ell_1}$, an equal distribution of energy over the scale notes results in a maximal chroma value of $g_n = 1/K_{sc}$ for each scale note with index n . Thus, the maximal value of S_q in Equation (5.20) is $(1/K_{sc})^{K_{sc}}$. We normalize with this factor and compute the final likelihoods as

$$P_{\text{scaletype}} := \frac{S_{\max}}{(1/K_{sc})^{K_{sc}}} \quad . \quad (5.25)$$

We obtain a maximum value of $P_{\text{scaletype}} = 1$ if all scale notes have equal energy and the off-scale notes have zero values:

$$g_q := \frac{1}{K_{sc}} \cdot T_q \quad (5.26)$$

with $q \in [0 : 11]$. For a graphical visualization of these analyses, we show the scale type likelihoods—indicated by the gray scale level—over time. We display the results for each frame from the beginning of the analysis window until the beginning of the next window. To compare the likelihood for different scale types, we use different template vectors \mathbf{T} . Note that we do not normalize the local histograms—in contrast to Section 5.2.4. Therefore, all scale type estimates may be high or low at the same time in principle.

We now want to present several examples for our scale type estimation algorithm. Non-diatonic scale types such as symmetrical scales have become important from the late romantic period on. In particular, composers from the impressionist period used pentatonic and whole tone scales, among others. In Figure 5.15, we show the analysis of C. Debussy’s prelude “Voiles.” We indicate the likelihoods $P_{\text{scaletype}}$ by different gray levels with a logarithmic color axis. In the first part until 1:50 min, the whole tone scale is dominating. This corresponds to the score, which only contains pitches from one whole tone scale for the first 41 measures.

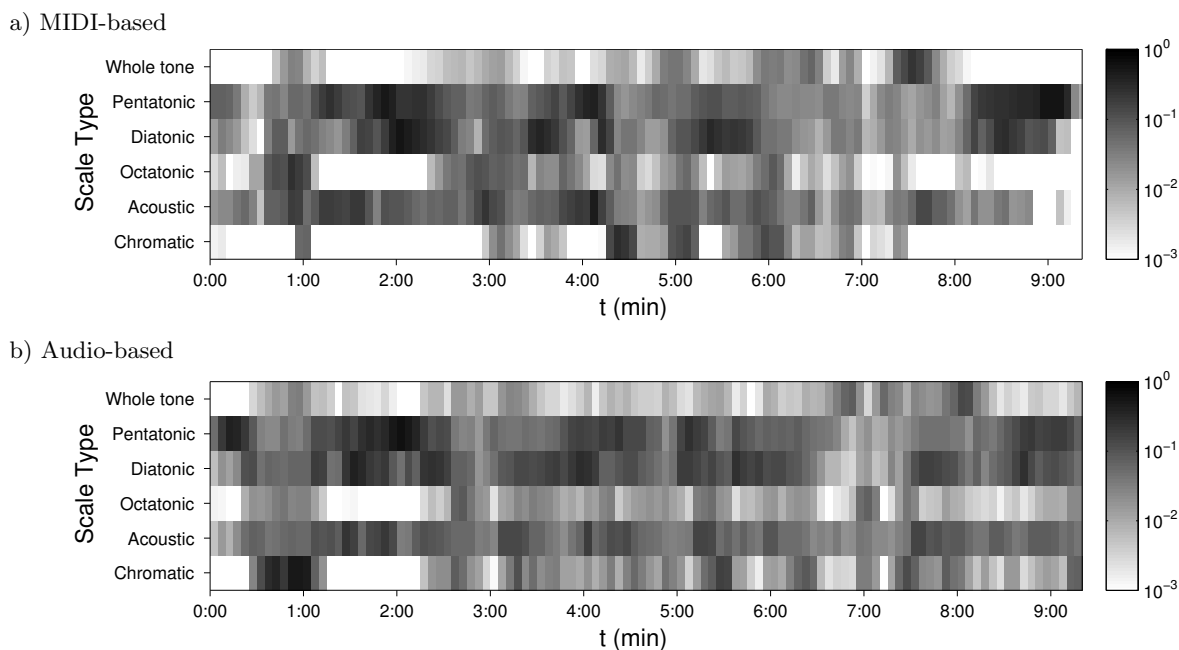


Figure 5.16. Scale type visualization of C. Debussy’s “La Mer.” This plot shows the first movement “De l’aube à midi sur la mer” from the orchestral piece. We compare analyses based on synthetic (not interpreted) MIDI data (a) and audio data (b) with $B_T = 200$ and $H_T = 50$. For the audio analysis, we use a recording played by the Belgian Radio and Television Philharmonic Orchestra under A. Rahbari (Naxos 1997).

In contrast, the middle part relies on a pentatonic scale. For the ending section, the music returns to the pitch class content of the whole tone scale. In the parts with dominating whole tone scale, we see some contributions to the likelihood for the acoustic scale as well. This is not very surprising, since the acoustic scale contains five out of the six notes of a whole tone scale. This close relationship—together with chroma artifacts stemming from upper harmonics or effects such as resonances in the piano—may lead to a non-zero likelihood for the acoustic scale. We observe a similar behavior comparing the pentatonic and the diatonic scales. Since the pentatonic scale pitches are a subset of the diatonic scale, small energy deviations in the “silent” chroma bands may produce a contribution to the diatonic scale likelihood—even if only the notes of a pentatonic scale are sounding.

Effects of this kind may cause even more problems when dealing with complex orchestral music, which exhibits a large variety of timbres. To investigate this, we show an analysis computed on a MIDI representation of C. Debussy’s orchestral piece “La Mer” and compare this analysis to the results of the audio-based method for the same piece (Figure 5.16). For the MIDI analysis, we weight the pitches with their velocity values and aggregate to pitch classes in order to build chroma-like features. On these features, we perform our analysis as described previously. Note that the time axes are not synchronized in a musically meaningful way so that the time positions only roughly relate to each other.

Comparing the results for the two representations (Figure 5.16 a, b), we observe a very similar structure. Looking at the details, we find some smaller deviations. In the ending sections (8:00–9:00 min), we find some “noisy” contributions to the likelihood of a chromatic scale for the audio analysis. In the beginning at about 0:30 min, we find more substantial differences. The reasons for the high likelihood of the chromatic scale in the audio analysis are not very clear since there is no indication in the score. Rather, an acoustic scale seems to be

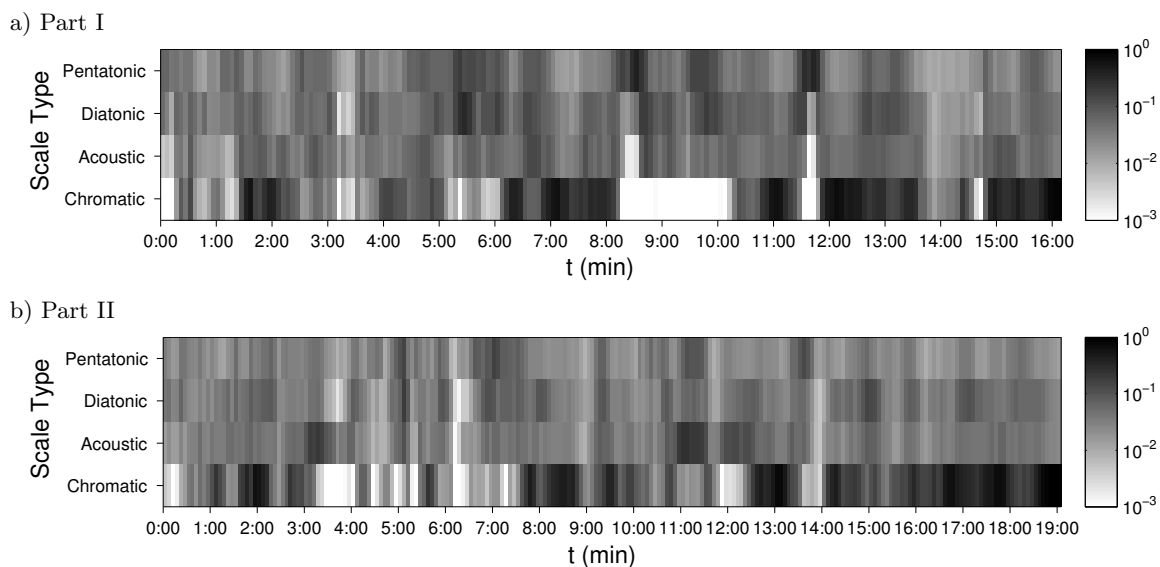


Figure 5.17. Scale type visualization of I. Stravinsky’s “Le Sacre du Printemps.” The upper plot shows the first part, the lower plot refers to the second part, $B_T = 200$, $H_T = 50$. We use a recording of the Belgian Radio and Television Philharmonic Orchestra, conducted by A. Rahbari (Naxos 1991).

present here. However, the audio-based method may have advantages as well. Gárdonyi [69] claims the horn motif in octaves to be an example for acoustic tonality. This motif first appears at about 1:45 (rehearsal letter¹¹ 3). Here, the audio-based analysis slightly better detects the presence of this scale. Moreover, we also notice the repetitions of that motif at about 3:00 min (letter 5) and around 4:00 min (short before letter 8) in the audio visualization. Nevertheless, these repetitions become more clear in the MIDI-based analysis. In general, we find a lot of pentatonic scales as well as some diatonic and acoustic scales. In contrast, there is almost no prominent whole tone scale. This may result from the fact that this scale appears simultaneously—as a kind of chord or “cluster”—less often than, for example, the pentatonic scale.

Next, we test our method on a piece containing atonal structures as well as parts dominated by percussion instruments. In Figure 5.17, we show an analysis of I. Stravinsky’s ballet music “Le Sacre du Printemps.” As we expected, we find high likelihoods for the chromatic scale in several sections of the piece. In particular, atonal and percussive phenomena may be present at the end of both parts. We find a contrasting section at the begin of the “Spring Rounds” movement (between 8:00 min and 10:00 min in the first part). Here, we find a pitch class selection related to the $E\flat$ dorian scale (level -5). This is one of the few sections of the piece that the composer notated with a key signature (5b). Indeed, we find highest likelihood for the diatonic scale here. For some sections, we observe indications for acoustic tonality. A weak example for such an observation is in the first part at 6:30 min (rehearsal letter 32)—in accordance to [69]. In the second part, there is a very clear indication for an acoustic scale at the beginning of the “Ritual Action of the Ancestors” at about 11:00 min (rehearsal letter 129). Here, we find a high likelihood for the acoustic scale, without ambiguities with other scales. The score analysis confirms this assumption. We see another indication for the acoustic scale in the second part at about 3:00 min (rehearsal letter 87). Analyzing the score

¹¹In most editions of this piece, the score has no measure numbers but rehearsal letters (in this case, numbers are used for this purpose). These markers serve to quickly identify important positions in the sheet music in order to clarify the structure and facilitate communication in rehearsals.

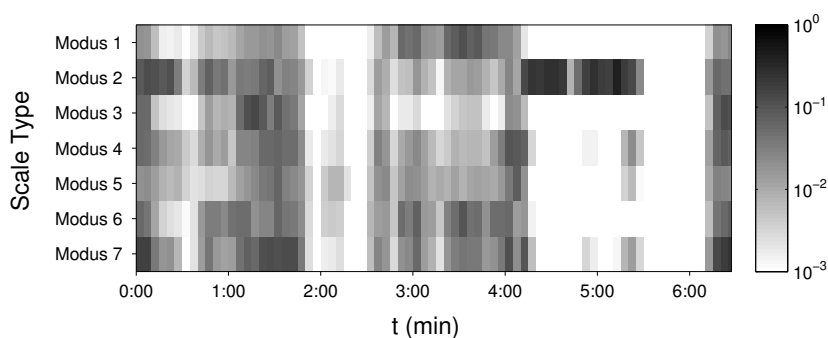


Figure 5.18. Visualization of O. Messiaen’s modes. For “La vierge et l’enfant,” No. 1 from O. Messiaen’s “La Nativité du Seigneur” for organ, we estimate the presence of the different modes, with $B_T = 150$, $H_T = 50$, based on a recording by D. G. Weir (Priority 1994).

leads to a similar result. The pitch classes of an acoustic scale based on $B\flat$ dominate this passage, with one additional pitch class ($D\flat$). Altogether, we see that this method can be helpful to get an overview over the tonal structure of large pieces. For pieces that combine different concepts of tonality, our approach can provide hints to particular tonal phenomena.

The scale type analysis presented in this section may be a suitable method for analyzing the music of O. Messiaen. In [159], he proposes a set of symmetrical scales called “modes of limited transposition”, which is crucial for his compositional approach. We already introduced some of these modes. The first mode corresponds to the whole tone scale and the second mode is the octatonic scale. The third mode relates to the hexatonic scale since it shows a periodicity in major third distance. The other three modes are periodic with respect to the tritone interval [159]. Here, we cannot give a full explanation of this theory. To illustrate the possibilities of our method for analyzing such music, we perform an analysis of an organ piece from “La Nativité du Seigneur,” shown in Figure 5.18. We find a clearly octatonic section in the last part of the piece between 4:00 min and 5:20 min (Modus 2). For the presence of other modes, we cannot see any clear indications. One reason for this may be the acoustic behavior of the organ. In this recording, aliquot registers—enhancing particular harmonics of the played pitches—have a strong influence on the sound. This may lead to deviations of the chroma features from the notated pitch classes. At the end of the piece (between 5:30 min and 6:00 min), none of the considered scale types seems to be present. Overall, the analysis of scale types is not satisfying for this piece even though several modes are present in the score. To investigate the problem of such analyses, further studies including MIDI representations of the pieces could be helpful.

5.2.6 Conclusion

In this section, we presented a novel approach for the computational analysis of audio data with respect to tonal and harmonic properties. The presented methods rely on chroma features grouped into analysis windows of variable length. We presented two post-processing methods inspired by several musicological theories. The first method locally estimates the likelihoods for the twelve diatonic scales over the course of a recording. We tested this method for music examples from several historical periods. Visualizing the results provides an overview of the modulation structure in a musically meaningful way—under the condition that the tonality of the music relies on diatonic scales. With the second analysis technique, we estimate the general scale type of the local tonal content. To do this, we match the chroma

vectors to binary templates of several scale types and extract the maximum likelihood for all transpositions of each scale model. We showed several examples from the 20th century where we identified fifth-based scale types (pentatonic, diatonic), symmetrical models (octatonic, hexatonic, whole tone scale), and acoustic tonality successfully. For atonal passages, we detected an enhanced likelihood for the chromatic scale.

If only a fraction of the scale notes is presented locally, the proposed analysis method might lead to problems and ambiguities. Therefore, the size and position and the analysis windows plays a crucial role. In the current system, the user has to manually adapt these parameters, which do not relate to musical time positions. Information about the musical time from automatic beat tracking or a manual annotation of the measure positions could improve analysis quality. This would also be helpful to link score positions to the analysis frames in an exact and reliable way. Furthermore, an adaptive approach could help to automatically improve clarity of visualizations by adjusting window parameters. Comparing the audio-based analysis to results computed on a MIDI representation of the same piece, we found only slight deviations pointing to a certain robustness against acoustical artifacts and noise. Altogether, both methods provide musically meaningful visualizations, which may help to get an overview of a piece's tonal shape.

6 Design of Tonal Features

In this chapter, we introduce further methods to automatically analyze the tonal content of music audio recordings. As opposed to Chapter 5—where we focused on the analysis of key and scale structures—, we present in Section 6.1 a procedure to estimate the occurrence of simultaneous interval and triad types from a chromagram. Section 6.2 comprises the description of several features for quantifying tonal complexity. We discuss these features’ characteristics by computing them for isolated chords. Furthermore, we visualize the feature values for selected movements of Beethoven’s piano sonatas in order to study their behavior in a realistic scenario. Most of the tonality measures proposed in this chapter serve as features for the classification experiments presented in Chapter 8.

6.1 Measuring Interval and Chord Categories

6.1.1 Introduction

Harmony mostly relates to the “vertical” way of combining musical tones. The analysis of harmony deals with musical constructs that sound simultaneously (**sonorities**), their quality, and their progression over time. The simplest form are harmonic intervals—two pitches sounding at the same time—since one can construct more complex sonorities by combining such intervals. A systematic way of interval-based analysis is the pitch class set theory [64,86] (compare Section 2.8.1). Furthermore, triads attained a particularly important role throughout Western music history so that some theorists consider triads as the basis of harmony rather than harmonic intervals [137].

In this section, we propose a method to quantify the occurrence of interval and triad categories. We compute these features on the basis of chroma representations with multiple temporal resolutions. The following considerations rely on [256] where we first introduced these features for the purpose of classifying musical styles. Here, we only describe the design of the features (Sections 6.1.2 and 6.1.3) and provide visualizations to illustrate their semantic meaning (Section 6.1.4). The classification experiments presented in [256] are topic of Chapter 8.

6.1.2 Extraction of Chroma Features

6.1.2.1 Chroma Feature Types and Enhancement

For describing the harmonic content of audio data without considering the details of timbre and instrumentation, chroma features were shown to be useful since they relate to the pitch class content of the music (compare Section 3.5.2). Scholars presented a number of different chroma feature extraction methods, which they evaluated with respect to different MIR tasks such as chord recognition (Section 3.5.3). One of the fundamental difficulties of the chroma representation is the influence of the partials: Each note played by an acoustical instrument generates a spectrum showing energy not only at the fundamental frequency but also at the integer multiples of this frequency. While the octave-related harmonics do not cause

problems in a chroma representation, harmonics corresponding to other pitches such as the upper fifths may lead to wrong musical interpretations. Several chroma extraction methods try to cope with this issue [76, 131, 147], as we discussed in Section 3.5.3. Exemplarily, we consider four different chroma computation techniques in this chapter:

- **CP.** Müller and Ewert [161, 170] present a chroma extraction method using a multirate pitch filter bank. We use the chroma pitch (CP) as published in the Chroma Toolbox package [165] as baseline representation. For the chroma computation, we consider pitch features in the piano range $p \in [21 : 108]$.
- **CLP.** For a chord recognition task, Jiang et al. [109] test several chroma features based on filter banks. They find significant improvement when using logarithmic compression before applying the octave mapping. We test the chroma logarithmic pitch (CLP) with compression parameter $\eta = 1000$ performing best in this evaluation.
- **EPCP.** Stein *et al.* [228] test a different chord matching algorithm. The enhanced pitch class profiles (EPCP) proposed by Lee [131] performed best in this study. This chroma feature is based on an iterative approach called harmonic product spectrum (HPS). We use three HPS iterations in the following studies.
- **NNLS.** In [147], Mauch and Dixon present an approximate transcription method using a non-negative least squares (NNLS) algorithm for chroma extraction. The authors use these features as input to a high-level model for chord transcription and evaluate on the MIREX Chord Detection task with good results (among the best systems in 2013 and 2014). They published their code as a Vamp plugin.¹

We compute all chroma feature representations with an initial feature rate of $f_{\text{feat}} = 10$ Hz using a hopsize of $H = 4410$ samples with an audio sampling rate of $f_s = 44.1$ kHz. We normalize the features to the Manhattan norm ℓ_1 in order to eliminate the influence of dynamics obtaining a chromagram \mathcal{C}^{ℓ_1} .

6.1.2.2 Multi-Scale Feature Smoothing

Since tonality is a hierarchical concept, tonal characteristics of music refer to various time scales. On a rough scale, the global key as well as local keys and modulations play an important role. Regarding a finer level, chords and their progressions provide more detailed information. Finally, considering the properties of melody and voice leading gives an insight into the relationship of the pitches to the underlying chords. These different layers of tonality are crucial for musical style recognition as well. Analyzing a piece of dodecaphonic music, we find a complex tonality making use of most of the chromatic pitches on a fine scale as well as on a global scale. A large-scale Romantic piece may look similarly complex globally due to numerous modulations while being built from rather simple constructs on a fine level.

Motivated by this, we consider different temporal resolutions for the computation of our features. To do this, we start with the chroma features introduced in Section 6.1.2 with a feature resolution of $f_{\text{feat}} = 10$ Hz. Then, we apply a feature smoothing to different resolutions. We use the approach proposed by Müller *et al.* [161, 167] for the CENS features with smoothing window length w and downsampling factor d given in frames as previously discussed (Section 3.5.5). After smoothing, we again normalize the feature frames using the

¹<http://isophonics.net/nnls-chroma>

Table 6.1. Chroma feature types for different time scales. Based on the initial chromagrams, we calculate several smoothed versions $[\text{Chroma}]_d^w$ specified by the parameters w (length of the smoothing window in frames) and d (downsampling factor).

<i>Feature type</i>				
<i>Temporal resolution</i>	$\text{CP}_{\text{global}}$	$\text{CLP}_{\text{global}}$	$\text{EPCP}_{\text{global}}$	$\text{NNLS}_{\text{global}}$
	CP_{100}^{200}	CLP_{100}^{200}	EPCP_{100}^{200}	NNLS_{100}^{200}
	CP_{20}^{100}	CLP_{20}^{100}	EPCP_{20}^{100}	NNLS_{20}^{100}
	CP_{10}^{20}	CLP_{10}^{20}	EPCP_{10}^{20}	NNLS_{10}^{20}
	CP_5^{10}	CLP_5^{10}	EPCP_5^{10}	NNLS_5^{10}
	CP_2^4	CLP_2^4	EPCP_2^4	NNLS_2^4
	CP_{local}	$\text{CLP}_{\text{local}}$	$\text{EPCP}_{\text{local}}$	$\text{NNLS}_{\text{local}}$

Table 6.2. Interval categories. For the categories IC1, \dots , IC6, we list the characteristic intervals and the associated interval distances in semitones.

<i>Category</i>	<i>Intervals</i>	Δ
IC1	m2 / M7	1 / 11
IC2	M2 / m7	2 / 10
IC3	m3 / M6	3 / 9
IC4	M3 / m6	4 / 8
IC5	P4 / P5	5 / 7
IC6	+4 / °5	6 / 6

ℓ_1 norm. Furthermore, we compute a global chroma histogram \mathbf{g}^{ℓ_1} for every feature type (denoted as $[\text{Chroma}]_{\text{global}}$). Together with the local features $[\text{Chroma}]_{\text{local}}$ (10 Hz), we obtain seven different temporal resolutions (see Table 6.1) for the experiments in [256].

6.1.3 Interval and Chord Features

Relying on chroma features such as the ones listed in Table 6.1, we compute semantic mid-level features describing the tonal content of the audio data at several time scales. Since we do not want our features to depend on the global or local key, these features have to be invariant under cyclic shifts of the chroma vector (musical transposition). With this requirement, the task relates to the analysis method of pitch class set theory (compare Section 2.8.1). This theory summarizes simultaneous sounds—harmonies or *sonorities*—to pitch class sets. These pitch class sets can be assigned to interval categories (IC) characterized by their predominant interval class. Since we are dealing with pitch classes here, we identify complementary intervals ending up with only six interval categories. In Table 2.3, we introduced the ICs and the construction of pitch class set prototypes for the categories. If we only consider harmonic intervals as sonorities (two-part pitch class sets), we obtain one pitch class set per category that describes an interval, its complementary and all related compound intervals (see Section 2.3 for the explanation of these terms). In Table 6.2, we list these intervals.

Based on this theory, Honigh and Bod [92, 93] performed classification and tonal analysis experiments on MIDI data, which showed that pitch class sets can be valuable style markers. We extend this approach to audio data using chroma features as basis. To this end, we use

simple binary templates modeling the interval and chord content of the music. Since we cannot discriminate between an interval and its complementary, the six interval categories in Table 6.2 are the only information left. For a fixed frame index $m \in [1 : M]$, every column of the chromagram forms a chroma vector $\mathbf{c} := \mathcal{C}^{\ell_1}(\cdot, m) \in \mathbb{R}^{12}$. For each of these vectors, we compute the likelihood for the joint appearance of two pitch classes that relate by the respective interval. To this end, we multiply their chroma values given by \mathbf{c} . For the feature $\Psi^{\text{IC}5}$ related to the intervals P4/P5, for example, we multiply the chroma value c_0 for pitch class C with the value c_5 for F ($q = 5$) forming an interval with distance $\Delta = 5$ semitones. Since we are interested in the type of the interval and not in the specific pitches, we equally weight all transpositions of this interval by summing over all cyclic shifts. We obtain the feature value

$$\Psi^{\text{IC}5}(\mathbf{c}) := \sum_{q=0}^{11} c_q \cdot c_{(q+5) \bmod 12}. \quad (6.1)$$

To generalize this expression, we use a binary template $\mathbf{T} := (T_0, \dots, T_{11})^T \in \mathbb{R}^{12}$:

$$\Psi^{\mathbf{T}}(\mathbf{c}) = \sum_{q=0}^{11} \left(\prod_{k=0}^{11} (c_{(q+k) \bmod 12})^{T_k} \right) \quad (6.2)$$

By suitably choosing \mathbf{T} , we can estimate the different interval categories:

$$\begin{aligned} \mathbf{T}^{\text{IC}1} &= (1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)^T \\ \mathbf{T}^{\text{IC}2} &= (1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0)^T \\ \mathbf{T}^{\text{IC}3} &= (1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0)^T \\ \mathbf{T}^{\text{IC}4} &= (1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0)^T \\ \mathbf{T}^{\text{IC}5} &= (1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)^T \\ \mathbf{T}^{\text{IC}6} &= (1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0)^T \end{aligned} \quad (6.3)$$

Using the template $\mathbf{T}^{\text{IC}5}$, we obtain the feature value $\Psi^{\text{IC}5}(\mathbf{c})$ as denoted in Equation (6.1).

We can easily extend this procedure to sets of three or more pitch classes. As the basic triads in Western tonality, we consider the triad types Major (M), Minor (m), Diminished ($^\circ$), and Augmented (+):

$$\begin{aligned} \mathbf{T}^{\text{M}} &= (1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0)^T \\ \mathbf{T}^{\text{m}} &= (1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0)^T \\ \mathbf{T}^{\circ} &= (1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0)^T \\ \mathbf{T}^+ &= (1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0)^T \end{aligned} \quad (6.4)$$

With this approach, we already include the triad inversions for the same triad type. Mathematically, this template matching strategy is identical to the scale type matching algorithm presented in Section 5.2.5. Here, we only use different templates describing intervals and chords rather than complete scales. In contrast to these simultaneous sounds, the concept of scales and (local) keys relates to larger sections of a musical piece. Therefore, a rather fine temporal resolution seems to be suitable for estimating intervals and chords whereas a more course time scale is required for estimating scale types.

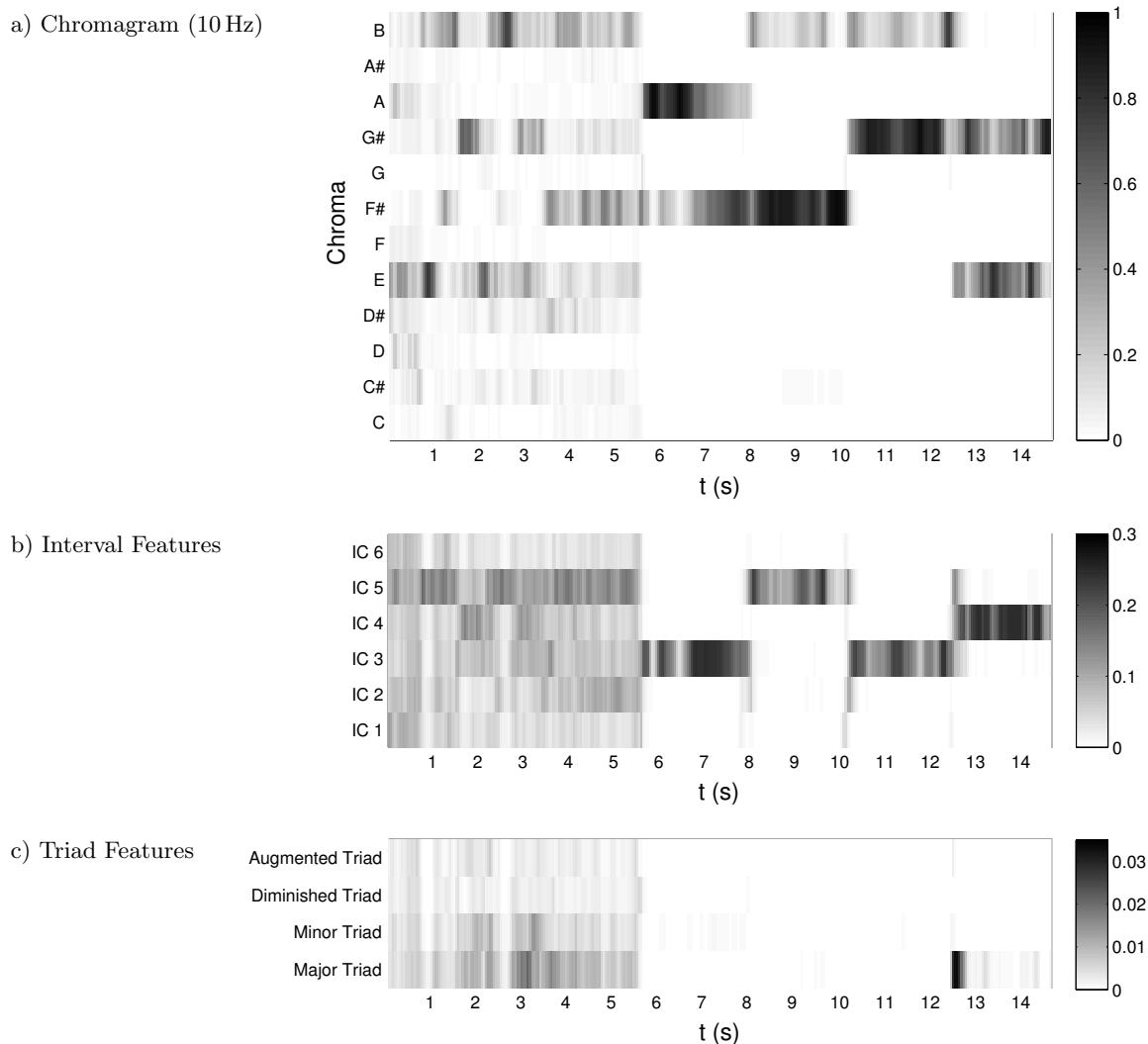


Figure 6.1. Template-based features for the “Fidelio” orchestra recording. For this excerpt, which we used as running example in Chapter 3, we show the normalized chromagram \mathcal{C}^{ℓ_1} of the first measures (upper plot (a)). Here, we use the CP chroma implementation in a resolution of 10 Hz (CP_{local}). The middle plot (b) shows the six interval features computed from the same chromagram. Plot (c) shows the feature values using the four triad templates. We encode the feature values by means of different gray levels. During the first five seconds, the full orchestra is playing in *forte*. The second part is a solo of two horns in *piano*.

In Chapter 8, we use all of the template-based features presented in this section $(\Psi^{\text{IC}1}, \dots, \Psi^{\text{IC}6}, \Psi^{\text{M}}, \Psi^{\text{m}}, \Psi^{\circ}, \Psi^+)^{\text{T}}$ as classification features, calculated for every chroma feature type of Table 6.1. To aggregate the frame-wise features Ψ for a whole piece, we calculate the mean and the standard deviation over the individual frames’ feature values.

6.1.4 Visualization Examples

To better understand the behavior of the features for real audio examples, we show some graphical examples for the features. In Figure 6.1, we present a short visualization for the “Fidelio” example from Chapter 3 based on the chromagram shown in Figure 3.14. Here, we use the CP_{local} feature to compute the different features. Looking at the interval estimates, we see two phases. For the first 6 s, the features show a considerable amount of noise stemming

from small non-zero values for most of the chroma values in (a). Nevertheless, the interval category IC5 exhibits larger values for this section. As a musical interpretation, we suppose that this results from the major triads EM and BM, which are written in the score here. The triad features confirm this assumption with a rather large value for the major triad type. In the second half of the example where only the horns are playing, the situation is different. Here, the chromagram is “cleaner” and concentrates the main energy in the pitch classes notated in the score. This leads to a high precision in the interval features that correctly indicate the interval sequence m3, P5, m3, and M3. Here, the triad features show low values. This is no surprise since only two voices are sounding at a time. However, we observe a strong value for the major triad at about 12.5s. This may arise from the overlap of the m3 interval G \sharp –B with the M3 interval E–G \sharp —maybe due to the reverb of the first sound.

To investigate the influence of the chroma feature implementation, we repeat the visualization of the interval features using different chroma features as input (Figure 6.2). The first plot (a) corresponds to the interval features of Figure 6.1 based on CP chroma. Using logarithmic compression (CLP) smoothes out the discontinuities to some degree leading to a nice visualization of the horn part. On the downside, this may flatten the chroma vectors too much. For the first seconds, we cannot see the enhanced feature values for IC5 and the third categories (IC3 and IC4) anymore. Using EPCP features, we see the enhancement of these categories in the first section. However, the interval estimates based on these features show a fluctuating behavior. For the second part, the third intervals are strongly suppressed compared to the other representations. This may arise from the low chroma values for some of the horn tones (compare Figure 3.15). Looking at the NNLS-based features, we find a small increase of noise compared to the CP-based representation. In general, these features seem to generate robust interval estimates. Here, the PC5 values in the beginning are in the same range as the horn intervals. Overall, the template-based features provide meaningful musical information but considerably depend on the quality of the underlying chroma features.

Finally, we examine the features’ dependency on the chroma smoothing step (Figure 6.3). We use four different smoothed versions of the CP chroma features as well as a chroma histogram over the whole audio excerpt. For small smoothing parameters (a–d), the features’ behavior only slightly deviates from using the initial CP chromagram. Using the global chroma histogram as input, the features change considerably. The intensity of the interval categories does not correspond to the statistics of the local features. The reason for this lies in the calculation of the histogram where all pitch class energies are summed up regardless of when they are sounding. Thus, we do not estimate the occurrence of locally simultaneous sounds with global features but describe properties of the overall pitch class statistics. However, these global properties may capture other meaningful information about tonality and musical style such as, for example, the relation of prominent local keys throughout a piece.

6.1.5 Conclusion

The method presented in this section serves to estimate the occurrence of simultaneous sonorities from audio recordings. The features rely on a normalized chromagram representation of the audio data. Furthermore, we account for different temporal scales by using several smoothed versions of the chromagram. We showed the features’ suitability for estimating harmonic interval and triad types by means of visualizations. Hereby, both the quality and robustness of the initial chroma features and the temporal smoothing showed considerable influence on the result. In the experiments presented in Chapters 7 and 8, we show the features’ efficiency for capturing stylistic properties of the music.

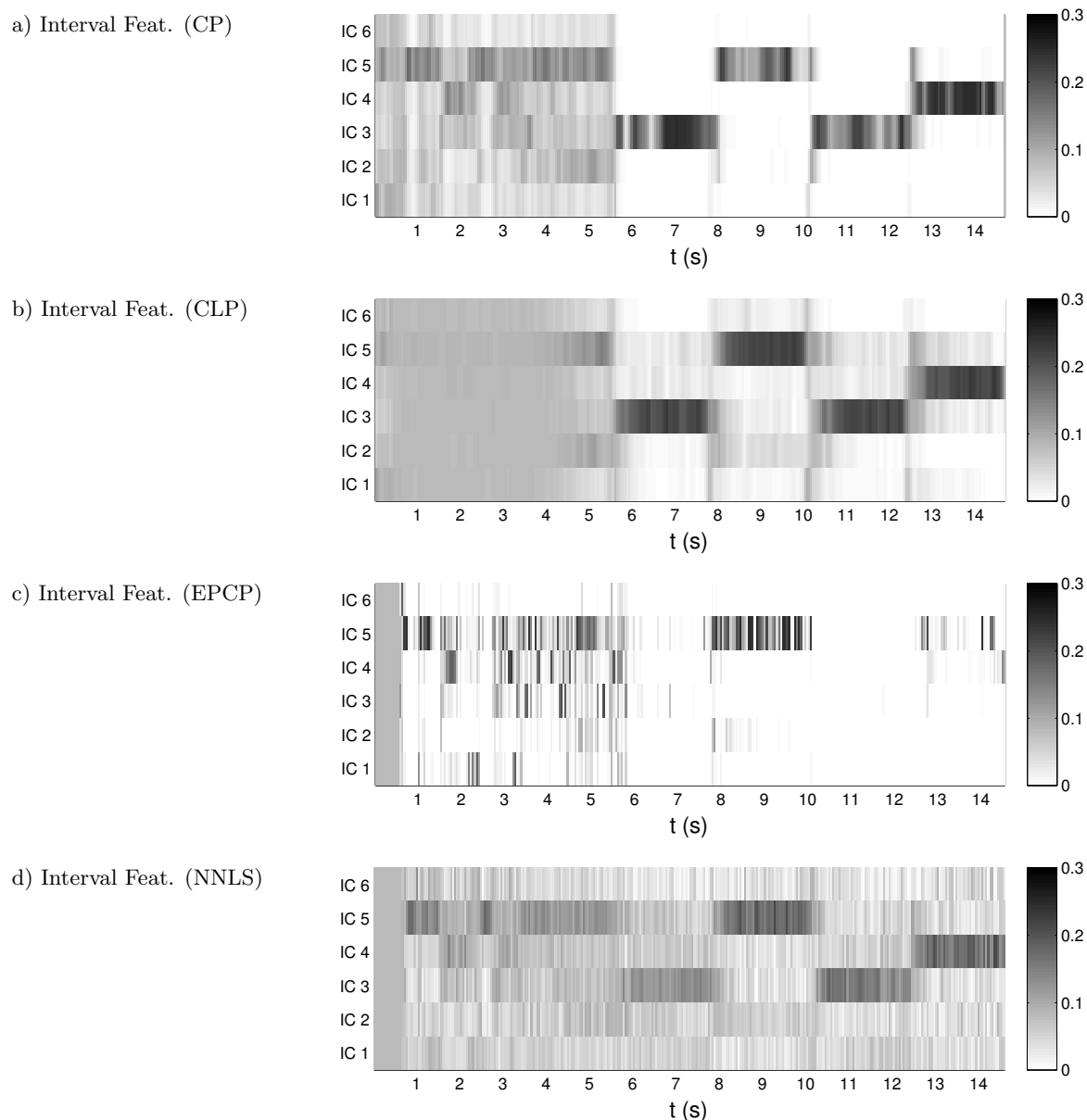


Figure 6.2. Interval features for the “Fidelio” example based on different chroma types. We compare the example of Figure 6.1 computed from different chroma implementations. In Figure 3.15, we showed these chromagrams for the same audio excerpt.

6.2 Quantifying Tonal Complexity

6.2.1 Introduction

In the previous section, we introduced features for quantifying the occurrence of specific tonal structures such as interval and chord types. Closely following our work published in [257], we now propose methods for describing a more abstract property of the music that we refer to as tonal complexity. We discussed the musical context of this notion in Section 2.9. To obtain a more precise definition, we compile in Section 6.2.2 a set of musical assumptions regarding

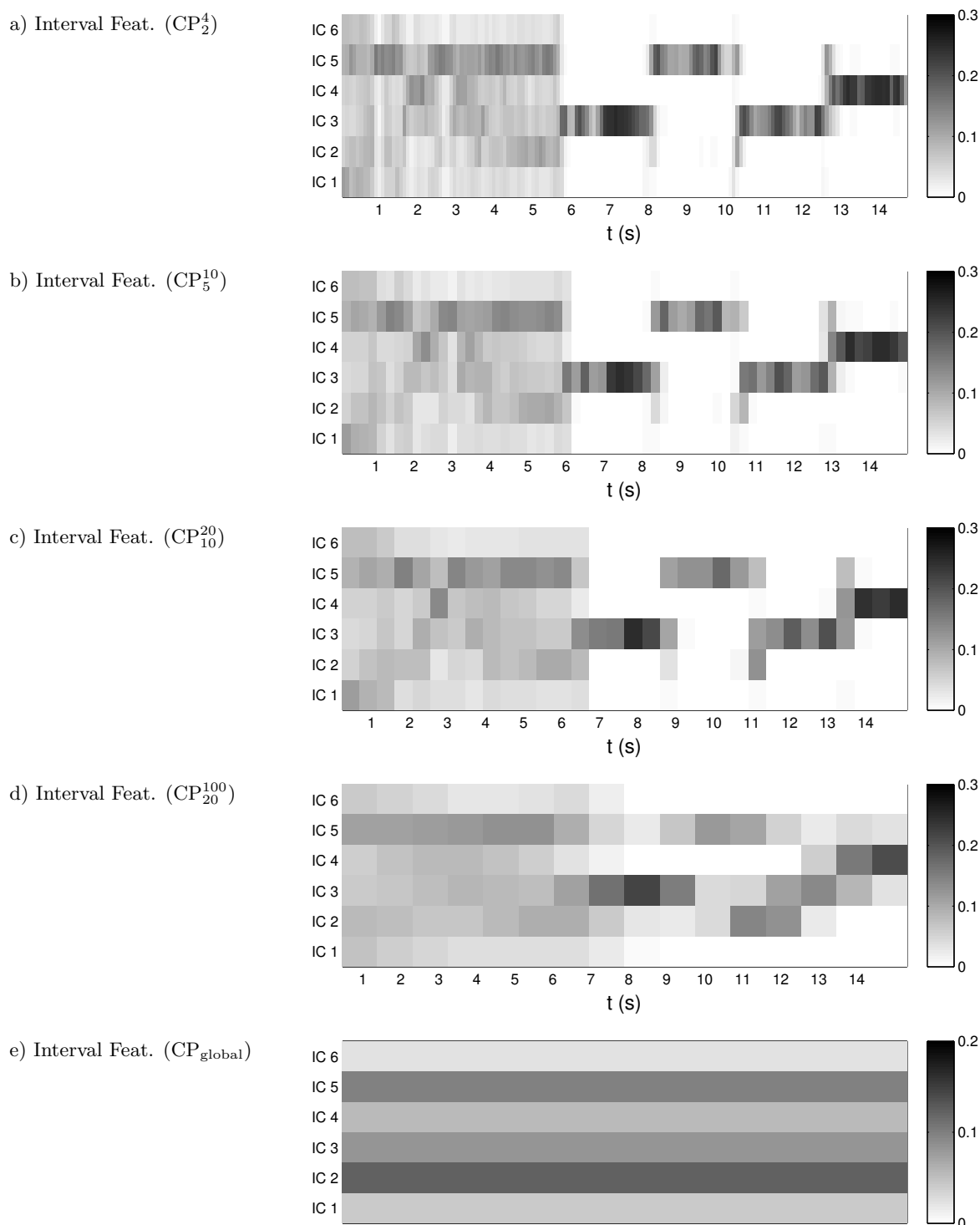


Figure 6.3. Interval features for the “Fidelio” example in different temporal resolutions. Here, we use different smoothed versions of the CP chroma for computing the interval features. For the lower plot (e), we calculate these features based on a chroma histogram over the whole example.

various temporal scales. In Section 6.2.3, we propose novel features for quantifying tonal complexity. The features rely on statistical measures calculated from chroma representations.

The characteristics of tonal complexity apply to different time scales. To illustrate this time scale dependence for the proposed features, we present hierarchical visualizations based on the previously introduced scape plot representation (Section 6.2.4). On a fine temporal level, tonal complexity relates to the characteristics of chords or scales. For example, in a modulating transition phase, we usually find more complex chords than at the beginning of a piece. To analyze such differences, we study the feature values for isolated chords (Section 6.2.4.1). Looking at a coarser level, the presence of modulations typically leads to an increase of tonal complexity. In the sonata form, for example, the development usually contains several modulations. To account for this property, we calculate the complexity features based on a coarse resolution of the chroma features. For evaluation of this coarse-scale complexity, we analyze selected movements of L. van Beethoven’s piano sonatas where we find higher complexity in the development parts (Section 6.2.4.2).

Beyond these experiments, we tested the benefit of our complexity features for classifying music styles [258]. We do not discuss these results in this section. Chapter 8 provides the results of our style classification experiments based on—among others—tonal complexity features. Nevertheless, we introduce in Section 6.2.3.2 all of the complexity measures and do not restrict ourselves to the three features discussed as examples in [257].

There are several attempts to approach similar concepts. Concerning symbolic music representations, Honing and Bod [92] test ideas from pitch class set theory to measure degrees of tonality. Kranenburg and Backer [242] use notions such as pitch entropy for style classification based on scores. Considering audio data, scholars proposed a few methods to quantify properties related to tonal complexity [150, 230]. They usually address sequential properties of harmony. We propose a different approach, accounting for the local pitch class distribution on various temporal scales. For a more profound literature survey, we refer to Section 4.5.

6.2.2 Musicological Implications

Assuming the existence of a musical dimension related to some kind of “tonal complexity,” we want to approach the meaning of this quantity by considering several musicological questions. From these questions, we define intuitive hypotheses that a tonal complexity measure should fulfill.

The quality of intervals and chords plays an important role to create stabilizing and destabilizing musical moments. Considering the simple cadence GM^7 – CM , the striving nature of the dominant seventh chord with the dissonant $^{\circ}5$ interval requires a resolution to a consonant chord. In late Romantic harmony, more complex resolution chords may appear as well. In that case, however, the previous chord often feels even more dissonant. Thus, tonal complexity on a chord level may relate to the dissonance grade of the local tonal content. A major chord suggests a more stable feeling to the listener than a diminished chord, a dominant seventh chord, or just this major chord while playing figurative nonchord tones.

On a coarser scale, the change of chords and their tonal relationships may influence complexity. This level refers to the scales representing the local pitch content, and the way these scales change. Chord changes within the pitch content of a diatonic scale do not sound very surprising, neither do chords from a neighboring key with only one or two new accidentals. In contrast, a CM chord followed by $F\sharp M$ without harmonic progression generates an abrupt change. Moreover, structural sections of a piece may show different complexity levels according to their role within musical form, thus constituting “areas of stability and instability in relation to a starting point” [130].

Motivated by these considerations, we want to find a measure—say Γ —that expresses some kind of complexity of the tonal content on various temporal levels:

- **Chord level.** Different chords or scales should show distinct complexity:

$$\Gamma(\text{“Complex chord”}) > \Gamma(\text{“Simple chord”}) \quad (6.5)$$

- **Fine structure.** The subparts of a sonata exposition should be different in complexity:

$$\Gamma(\text{“Transition phase”}) > \Gamma(\text{“Theme”}) \quad (6.6)$$

- **Coarse structure.** The parts of a sonata form movement should show specific trends in complexity:

$$\Gamma(\text{“Development”}) > \Gamma(\text{“Exposition”}) \quad (6.7)$$

- **Cross-work.** Considering the oeuvre of one composer, we expect the late works to be more complex than the early ones:²

$$\Gamma(\text{“Late sonata”}) > \Gamma(\text{“Early sonata”}) \quad (6.8)$$

- **Cross-composer.** On a cross-composer level, we assume stylistic trends. The historical periods may exhibit different levels of complexity:

$$\Gamma(\text{“Romantic”}) > \Gamma(\text{“Classical”}) \quad (6.9)$$

We are conscious of the limitations of these rather simplistic assumptions and use them only as a guiding principle for testing certain tendencies. For verifying some of the hypotheses, we may need perceptual studies and listening tests, others require a closer look at the musical scores and a detailed view on musical styles. In Section 6.2.3.2, we propose several mathematical realizations of such measure Γ on the basis of pitch class distributions.

6.2.3 Proposed Method

6.2.3.1 Extraction of Chroma Features

For an appropriate description of tonality, we want the complexity features to be invariant against timbral variations. For example, an orchestra chord should obtain a similar value as the same chord played on a piano. Thus, we build our systems on chroma features, which were shown to capture tonal information and to be invariant against timbral variations to a large extent (compare Section 3.5.2).

In the following, $\mathbf{c} := (c_0, c_1, \dots, c_{11})^T$ denotes a chroma vector as introduced in Section 3.5.2. For the chroma extraction, we employ the CLP chroma implementation from the Chroma Toolbox package [165]. We use a feature resolution of 10 Hz and normalize the features column-wise such that $\ell_1(\mathbf{c}) = 1$. In the following, \mathbf{c} may refer to the columns

²This assumption may only be true for some composers (such as A. Schönberg). For others, we may find the opposite kind of evolution (K. Penderecki). In both cases, we find some kind of “change” or “evolution” over the lifetime of the composer. For this reason, we may consider this as an analysis scenario rather than a guideline for feature design. In Section 6.2.4.2, we study this evolution for L. van Beethoven’s piano sonatas.

$m \in [1 : M]$ of a local chromagram $\mathcal{C}^{\ell_1}(\cdot, m)$ or to a chroma histogram \mathbf{g}^{ℓ_1} computed over several local vectors.

In this section, we introduce some basic concepts for quantifying tonal complexity. Thus, we do not optimize the chroma extraction by considering higher partials or other enhancement methods. For further improvements, it may be necessary to consider more advanced chroma computation methods such as the ones presented in Section 3.5.3. In order to account for the logarithmic behavior of loudness perception, we apply a logarithmic compression before the normalization step (Section 3.5.3.3). Inspired by Jiang *et al.* [109], we choose the parameter $\eta = 100$ for our experiments.

6.2.3.2 Complexity Features

Motivated by the considerations presented in Section 2.9, we want to find a measure—say Γ —that expresses the complexity of the (local) tonal content. To this end, we now propose several statistical measures calculated on a chroma vector. We want the feature values to increase for growing tonal complexity and scale to unit range:

$$0 \leq \Gamma \leq 1. \quad (6.10)$$

The basic idea of all these features is to compute a measure for the flatness of the chroma distribution. This is motivated by the following considerations. On a fine level, the simplest tonal item may be an isolated musical note represented by a Dirac-like (“sparse”) pitch class distribution

$$\mathbf{c}^{\text{sparse}} := (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)^T. \quad (6.11)$$

To this vector, we want to assign the lowest complexity value $\Gamma(\mathbf{c}^{\text{sparse}}) = 0$. Furthermore, a sparser chromagram describing, for example, a diatonic scale should obtain a smaller complexity value than an equal (“flat”) distribution

$$\mathbf{c}^{\text{flat}} := \frac{1}{12} \cdot (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)^T. \quad (6.12)$$

The latter case—where all twelve pitch classes have the same energy—could occur for dodecaphonic music. We want to rate this flat distribution with the highest complexity value: $\Gamma(\mathbf{c}^{\text{flat}}) = 1$. Following these guidelines, we present a number of features for capturing such characteristics. Though not all features are fulfilling all of the hypotheses from Section 6.2.2, the individual features may contribute to model different aspects of tonal complexity.

(1) Sum of chroma differences: To account for harmonic similarity of pitch classes, we resort the chroma values to an ordering of P5 intervals (7 semitones) $\mathbf{c}^{\text{fifth}} := (c_0^{\text{fifth}}, \dots, c_{11}^{\text{fifth}})^T$:

$$c_q^{\text{fifth}} = c_{(q \cdot 7 \bmod 12)} \quad (6.13)$$

with $q \in [0 : 11]$. Then, we compute the absolute differences between all neighboring chroma values:

$$\tilde{\Gamma}_{\text{Diff}}(\mathbf{c}) := \sum_{q=0}^{11} |c_{(q+1) \bmod 12}^{\text{fifth}} - c_q^{\text{fifth}}|. \quad (6.14)$$

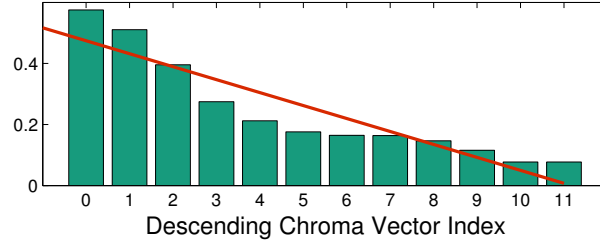


Figure 6.4. Linear fit to descending chroma values. The chroma values correspond to the global chroma histogram of the “Fidelio” example, re-ordered to a descending series.

Since $\tilde{\Gamma}_{\text{Diff}}(\mathbf{c}^{\text{flat}}) = 0$ and $\tilde{\Gamma}_{\text{Diff}}(\mathbf{c}^{\text{sparse}}) = 2$, we rescale this feature with $\gamma_1 := 2$:

$$\Gamma_{\text{Diff}}(\mathbf{c}) := 1 - \frac{\tilde{\Gamma}_{\text{Diff}}(\mathbf{c})}{\gamma_1} \quad (6.15)$$

(2) Standard deviation of the chroma vector:

$$\tilde{\Gamma}_{\text{Std}}(\mathbf{c}) := \sqrt{\frac{1}{11} \sum_{q=0}^{11} \left(c_q - \frac{1}{12} \sum_{k=0}^{11} c_k \right)^2} \quad (6.16)$$

The standard deviation reaches its maximum for a sparse distribution $\gamma_2 := \tilde{\Gamma}_{\text{Std}}(\mathbf{c}^{\text{sparse}}) = 1/\sqrt{12} \approx 0.29$ so that we calculate the rescaled feature in the following way:

$$\Gamma_{\text{Std}}(\mathbf{c}) := 1 - \frac{\tilde{\Gamma}_{\text{Std}}(\mathbf{c})}{\gamma_2} \quad (6.17)$$

(3) Negative slope of a linear function: We re-order the chroma vector entries to a descending series

$$\mathbf{c}^{\text{descend}} := (\max_q c_q, \dots, \min_q c_q). \quad (6.18)$$

To measure the flatness, we apply linear regression assuming c_i^{descend} being dependent on the index i (see Figure 6.4). The slope $\lambda(\mathbf{c}^{\text{descend}})$ of the line that best fits $\mathbf{c}^{\text{descend}}$ in a least squares sense serves as feature value. For a sparse chroma vector, the fitted line has a slope of $\lambda(\mathbf{c}^{\text{sparse}}) \approx -0.039$. Hence, we rescale this feature with $\gamma_3 = 0.039 \approx \lambda(\mathbf{c}^{\text{sparse}})$:

$$\Gamma_{\text{Slope}}(\mathbf{c}) := 1 - \frac{|\lambda(\mathbf{c}^{\text{descend}})|}{\gamma_3} \quad (6.19)$$

(4) Shannon entropy of the chroma vector, after re-normalization to $\ell_1(\mathbf{c}) = 1$:

$$\Gamma_{\text{Entr}}(\mathbf{c}) := -\frac{1}{\log_2(12)} \left(\sum_{q=0}^{11} c_q \cdot \log_2(c_q) \right) \quad (6.20)$$

With the re-normalization, the boundary conditions $\Gamma_{\text{Entr}}(\mathbf{c}^{\text{flat}}) = 1$ and $\Gamma_{\text{Entr}}(\mathbf{c}^{\text{sparse}}) = 0$ are fulfilled

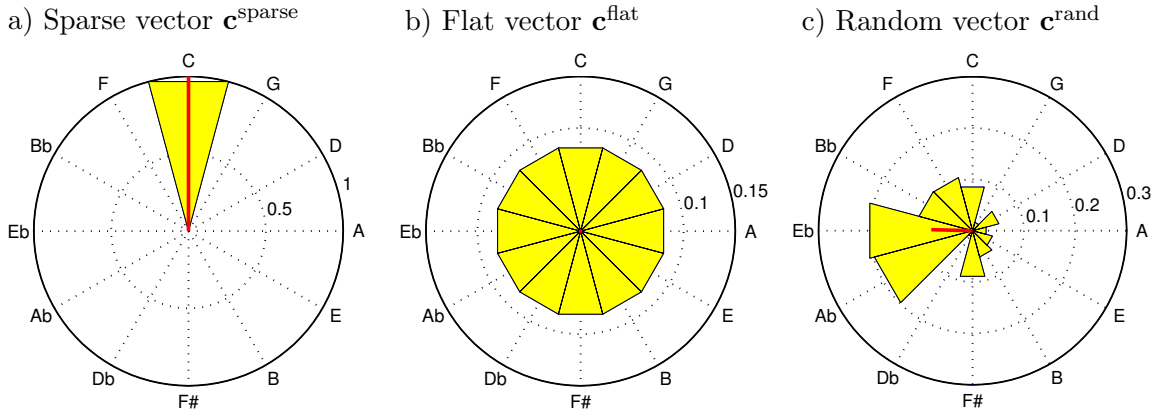


Figure 6.5. Circular interpretation of chroma vectors. The length of the yellow bars corresponds to the chroma vector entries $c_q^{\ell_1}$ with $q \in [0 : 11]$. We equally distribute the twelve chroma values over the circle. The red line indicates the resultant vector. For a sparse chroma vector $\mathbf{c}^{\text{sparse}}$, the resultant vector has length 1 (Figure (a)). A flat vector \mathbf{c}^{flat} obtains length 0 (Figure (b)). In Figure (c), we illustrate this principle for a random-like chroma vector.

(5) Non-Sparseness feature based on the relationship of ℓ_1 - and ℓ_2 -norm [96], inverted by subtraction from 1:

$$\Gamma_{\text{Sparse}}(\mathbf{c}) := 1 - \frac{\sqrt{12} - \|\mathbf{c}\|_1 / \|\mathbf{c}\|_2}{\sqrt{12} - 1} \quad (6.21)$$

This feature naturally lies between 0 and 1.

(6) Flatness measure describing the relation between the geometric and the arithmetic mean [184]:

$$\Gamma_{\text{Flat}}(\mathbf{c}) := \frac{\left(\prod_{q=0}^{11} c_q\right)^{1/12}}{\frac{1}{12} \sum_{q=0}^{11} c_q} \quad (6.22)$$

The flatness has values between 0 and 1.

(7) Angular deviation of the fifth-ordered chroma vector: We re-sort the chroma values according to Equation (6.13) obtaining a circular distribution of the pitch class energies—similar to the circle of fifths but now referring to pitch classes instead of musical keys. From this, we calculate the length of the mean resultant vector

$$r_{\text{fifth}}(\mathbf{c}) = \left| \sum_{q=0}^{11} c_q^{\text{fifth}} \exp\left(\frac{2\pi i q}{12}\right) \right|. \quad (6.23)$$

In Figure 6.5, we illustrate this circular interpretation together with the resultant vector for three different chroma vectors. From the resultant vector, we obtain the angular deviation via

$$\Gamma_{\text{Fifth}}(\mathbf{c}) := \sqrt{1 - r_{\text{fifth}}(\mathbf{c})}. \quad (6.24)$$

This way, Γ_{Fifth} describes the spread of the pitch classes. A short resultant vector—corresponding to a flat chroma vector—results in a high complexity value Γ_{Fifth} .

All of the proposed features take values between 0 and 1 and fulfill the conditions $\Gamma(\mathbf{c}^{\text{sparse}}) = 0$ and $\Gamma(\mathbf{c}^{\text{flat}}) = 1$. The features Γ_{Diff} and Γ_{Fifth} respect the ordering of the chroma entries and penalize distant relations in a perfect fifth sense. The remaining features

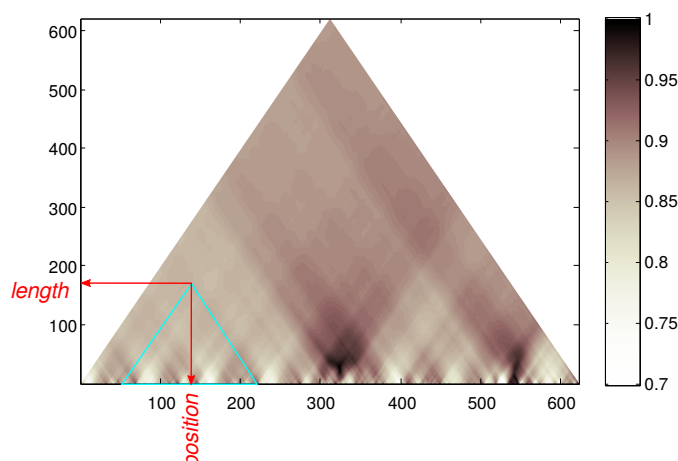


Figure 6.6. Example for a scape plot visualization. The horizontal axis gives the time position in seconds, whereas the vertical axis refers to the length of the segment. We indicate the feature values by different gray levels.

are invariant against permutation of the chroma vector entries. With this set of features, we consider several flatness-related aspects of a chroma vector. In Section 6.2.4.1, we discuss the individual features’ properties for single notes, chords, and scales.

6.2.3.3 Scale Dependence

The measurement of complexity crucially depends on the time scale of the observation. On a chromagram with fine resolution, the measures give an estimate of the complexity of chords and local scales. Regarding coarser levels, we calculate the complexity of several bars or a whole section. Using a chroma histogram as input, the complexity value refers to the full movement.

To examine the dependence of our proposed features, we visualize them hierarchically on different time scales, using the scape plot technique by Sapp [209,210]. With this techniques, we visualize different time scales in one plot. Figure 6.6 shows such a scape plot. The horizontal axis indicates the time position of the analysis window (mean). The vertical axis indicates the window’s length. The colors encode the feature value for every point. For example, the highest point gives the value for the complete recording. In the lowest row, we find the values for the local chroma vectors.

6.2.4 Evaluation

6.2.4.1 Chord Study

To better understand the proposed features, we analyze their behavior for different local items of tonality such as single pitches, intervals, chords, and scales. First, we do this for synthetic versions of these items and calculate the feature values for idealized binary templates. For example, the major chord template is $\mathbf{c}^{\text{Major}} = \tilde{\mathbf{c}}^{\text{Major}} / \ell_1(\tilde{\mathbf{c}}^{\text{Major}})$ with

$$\tilde{\mathbf{c}}^{\text{Major}} = (1, \varepsilon, \varepsilon, \varepsilon, 1, \varepsilon, \varepsilon, 1, \varepsilon, \varepsilon, \varepsilon, \varepsilon)^T. \quad (6.25)$$

To avoid degeneration in formulas due to zero entries, we use a small value for the silent pitch classes. We compare the results for $\varepsilon = 0$ and $\varepsilon = 0.05$ in order to estimate the consequences

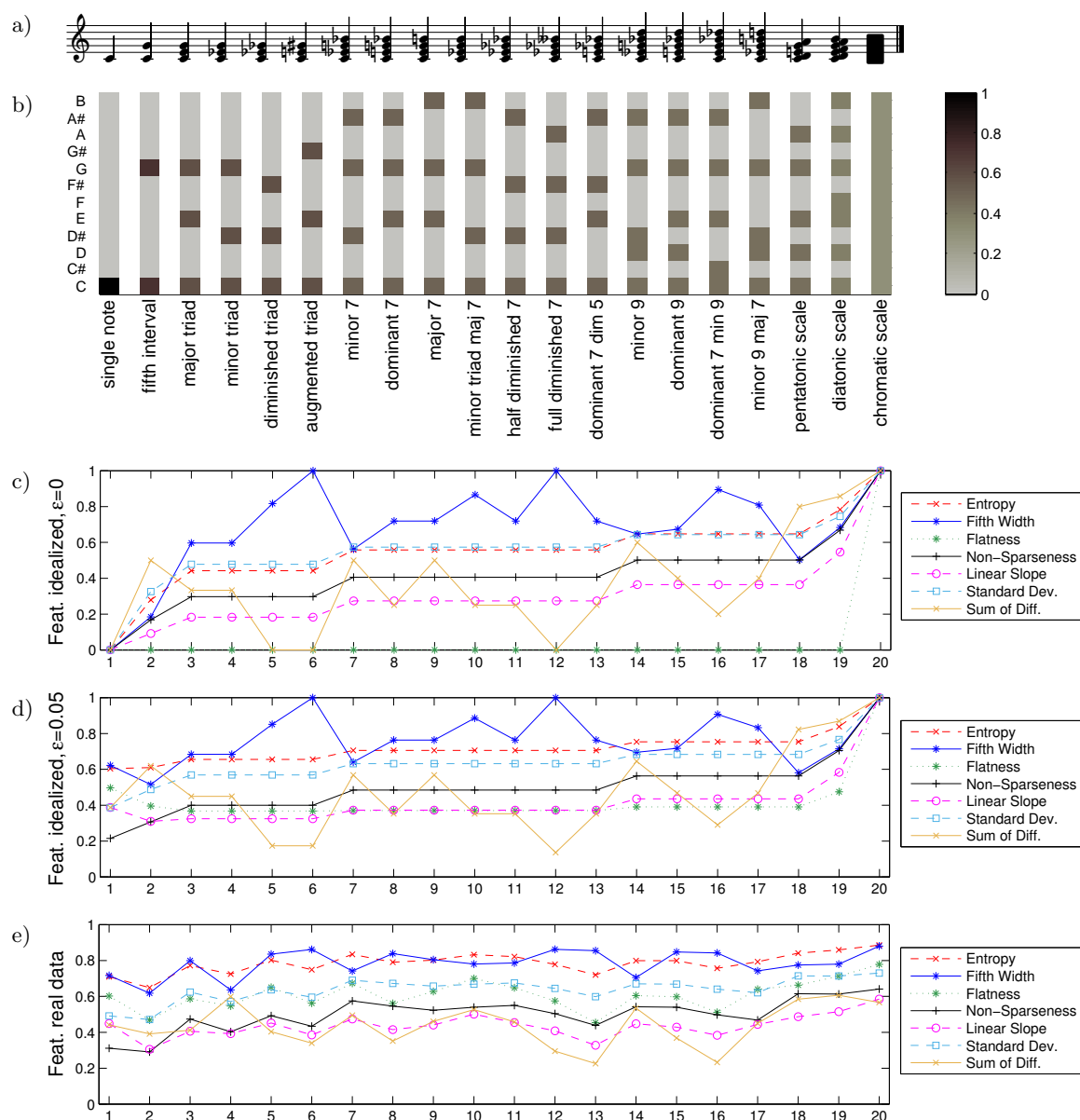


Figure 6.7. Complexity feature values for different tonal items. In (a), we display the musical notations of the sonorities. The next plot (b) illustrates the idealized chroma templates of the items (with $\varepsilon = 0.05$). Figure (c) shows the values for the ideal templates with $\varepsilon = 0$, Figure (d) for $\varepsilon = 0.05$. In the lowest part (e), we visualize the feature values for the recorded piano chords using CLP chroma with a compression parameter of $\eta = 100$ [110].

of this effect. Second, we analyze real audio recordings of the same chords, played on a piano for approximately 3s. We calculate chroma histograms over this short time span and use them as input for computing the complexity features. As tonal items, we consider a single pitch, a fifth interval, the four basic triads, seven types of seventh chords, four types of ninth chords, and three scales (pentatonic, diatonic, and chromatic). The results of this study are shown in Figure 6.7.

First, let us start with the results for ideal chord templates. With $\varepsilon = 0$ (Figure 6.7 c), all features assume the value $\Gamma = 0$ for the single pitch (No. 1). For Γ_{Entr} , Γ_{Sparse} , Γ_{Slope} , and Γ_{Std} , the feature values increase monotonically with growing number of notes. For these four features, a seventh chord obtains a higher complexity value than a triad. They correspond to some “degree of polyphony” of the local chords. In contrast, Γ_{Fifth} and Γ_{Diff} account for the ordering of the pitches. For example, Γ_{Fifth} obtains a higher value for a diminished triad (No. 5) than for a major triad value (No. 3) since the diminished triad has a larger spread on a perfect fifth axis. Symmetric divisions of the octave such as the augmented triad (No. 6) and the full-diminished seventh chord (No. 12) obtain maximal Γ_{Fifth} values. In contrast, the pentatonic scale (No. 18) with five pitches has a relatively small Γ_{Fifth} value since all pitches related by perfect fifths. Γ_{Diff} especially reacts on the number of perfect fifth intervals inside a chord. So, the augmented triad (No. 6) or the diminished seventh chord (No. 12) obtain $\Gamma_{\text{Diff}} = 0$ since they show no fifth interval. In contrast, the pentatonic scale (No. 18) obtains a high value. Γ_{Flat} is very sensitive to degradations since one zero value in the chroma vector already leads to $\Gamma_{\text{Flat}} = 0$.

We observe a different behavior with $\varepsilon = 0.05$. In this case, Γ_{Flat} does not assume zero values but rather reacts on the number of notes. Similarly, the chords with $\Gamma_{\text{Diff}} = 0$ obtain a higher value now. Beyond these effects, only slight changes appear. Interestingly, the fifth interval (No. 2) obtains a smaller value than the single note (No. 1) when having a non-zero ε . For the chromatic scale (No. 20), both configurations lead to $\Gamma = 1$.

For the recorded chords, differences in intensity appear in the chroma vector, although the chords are played with approximately equal loudness. The features react on these variations so that the above mentioned observations are less clear for the real piano chords. Γ_{Flat} turned out particularly sensitive to this effect. To improve the robustness of the features, more elaborate chroma features with respect to timbre invariance could be useful (compare Section 3.5.3). In our experiments, logarithmic compression in the chroma computation (compare Section 3.5.3.3) led to noticeable improvements for the real chords and, thus, seems to be an important step for computing robust complexity features.

6.2.4.2 Study on L. van Beethoven’s Piano Sonatas

As the next example, we want to study the piano sonatas of L. van Beethoven’s in a recording by D. Barenboim. Even though they are not standard sonata examples of their time but full of surprising ideas and changes, we can observe some general trends. In the upper part of Figure 6.8, we show three scape plots as introduced in Section 6.2.3.3. To compute the plots, we average the original 10 Hz chroma features at different window sizes. The horizontal axis gives the position of the segment in seconds, the vertical axis corresponds to the length of the segment. The lowest row describes a local level, the triangle’s top gives a single value for the full recording. We encode the feature value for the respective segment by the color’s darkness. For all three movements, we see a dark region indicating high complexity for the development phases. We can well recognize the similarity between the exposition and its repetition. Regarding the fine structure, we see bright phases corresponding to the themes and dark phases describing the higher complexity in the transition phases. In the development, the global complexity is always high—in contrast to the local one. This may arise from development parts without complex chords but with complex modulations—covering distant keys within a short segment. Looking at the development of the “Appassionata” Op. 57 (Figure 6.8 b), a modulating phase is followed by a long segment in A^b major indicated by a white section that starts at 240 s.

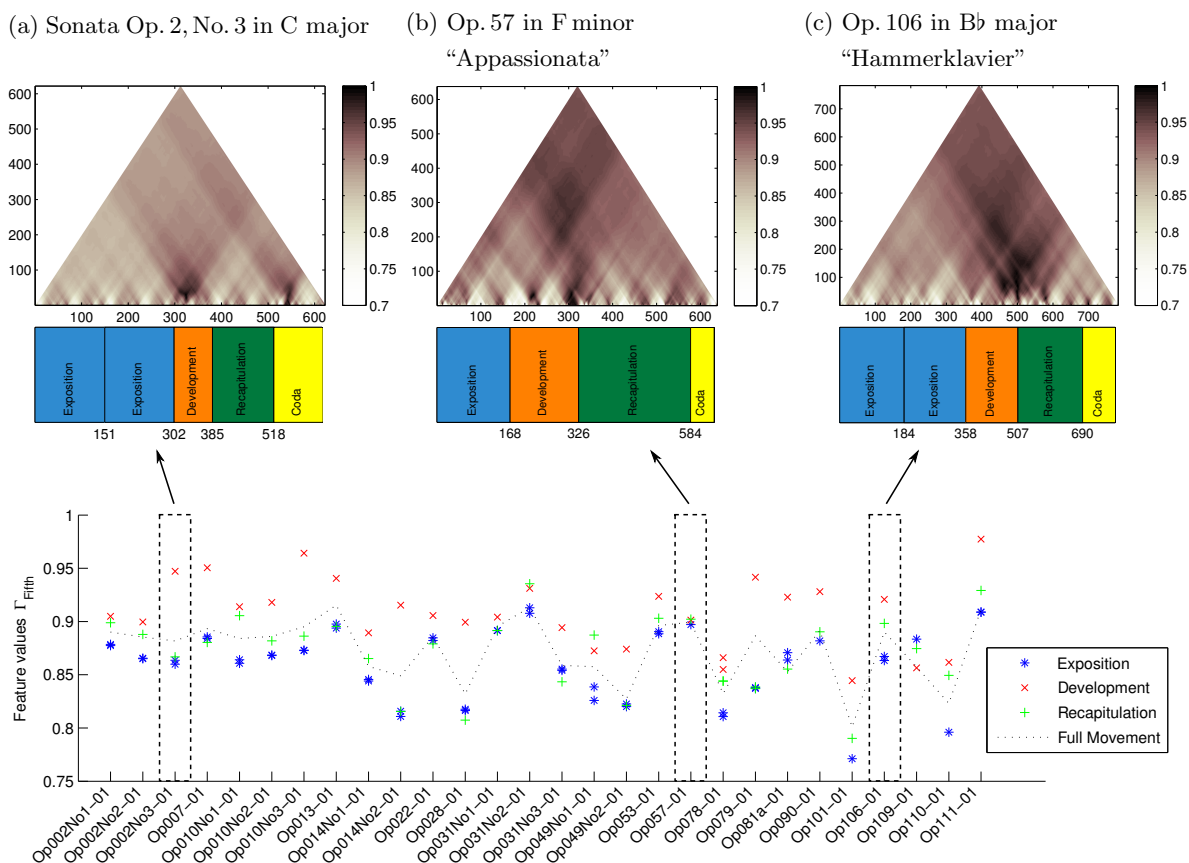


Figure 6.8. Tonal complexity analysis for selected movements from Beethoven’s sonatas. In the upper row, we show scape plots using Γ_{Fifth} for the first movements of three selected sonatas. For an overview, we display Γ_{Fifth} for the all first movements of L. van Beethoven’s sonatas that are in sonata form (lower figure). We calculate the features for the individual parts on a 100s level (0.01 Hz) and average.

To test the coarse structure hypothesis (Equation (6.7)), we plot the average Γ_{Fifth} values for the main parts of the 28 head movements composed in sonata form (Figure 6.8, lower part). The complexity in the development phase is always highest, with four exceptions. One case is the sonata Op. 109, where the development shows almost no modulations. Rather, the movement consists of alternating parts with similar harmonic structure. In the G minor sonata Op. 49, No. 1, the development contains a long stable $E\flat$ major part and, thus, does not obtain a high complexity score. In contrast, the recapitulation of this movement yields a high Γ_{Fifth} value—clearly higher than the exposition. One reason for this observation may be the local key structure of the sonata form in minor keys. In the exposition, the second theme usually stands in the relative major key and, thus, contains mainly one diatonic scale. In the recapitulation, this part is transposed to the global key (minor key), which includes pitches from the harmonic and melodic minor scales, leading to a higher complexity. We observe a similar effect for other movements in minor keys (Op. 2, No. 1 or Op. 10, No. 1). In general, the recapitulation seems to be slightly more complex than the exposition. This may arise due to additional harmonic effects, which serve to vitalize the non-modulating repetition of the familiar exposition material.

In Op. 79, we find a contrasting scenario with a stable exposition section followed by a strongly modulating development, which touches the local keys E major, C major, C minor, $E\flat$ major, and changes back to G major. In future work, it could be useful to combine

the discussion of further details with analyses of modulations such as the ones presented in Section 5.2 or [110].

Regarding global complexity, the hypothesis in Equation (6.8) assuming increasing values over the course of a composer's lifetime does not hold. The scores for the late works change substantially—a hint to high individuality of the compositions—in contrast to the early sonatas, which show a similar complexity structure among each other. Within the late sonatas, we find the most extreme values—the light and tonally constant Op. 101 in E major in contrast to the last sonata Op. 111 in C minor with complex harmony full of dissonances and a polyphonic development. Trusting in our features, however, we cannot confirm a general trend towards higher complexity with increasing composition time. This observation is consistent with the results of [190].

6.2.5 Conclusion

In this section, we presented novel features for quantifying the complexity of music regarding tonality. We compiled a set of assumptions to define requirements for the features' characteristics. In a study with ideal chord templates as well as recorded piano chords, we tested these assumption on a fine temporal level. Hierarchical visualizations of complexity values for movements of Beethoven's sonatas show the features' capability to capture the structure of the sonata form. Development parts and transition phases between themes show a higher complexity, in general. We could verify this behavior for most of the first movements in L. van Beethoven's piano sonatas.

7 Clustering and Analysis of Musical Styles

In the Chapters 5 and 6, we presented different types of features for capturing tonal characteristics of audio recordings. In several case studies, we showed these features’ behavior for individual pieces, segments, or isolated chords. We now want to analyze such kind of descriptors for analyzing databases of Western classical music with respect to style characteristics. Hereby, we make use of methods from the fields of data analysis and machine learning as presented in Section 3.6. In this chapter, we focus on unsupervised methods in order to get insights into the structure of our corpus with respect to stylistically similar pieces—*without incorporating primary assumptions* about historical or stylistic periods. As opposed to this, Chapter 8 deals with the automatic classification of pieces into *pre-defined style categories*.

In Section 7.1, we describe the dataset that we compiled for our analyses. Section 7.2 presents a method for mapping the feature values of individual pieces onto a historical time axis. Finally, we perform unsupervised clustering experiments (Section 7.3) in order to automatically group pieces, years, or composers on the basis of our features.

7.1 Dataset

In this thesis, we are interested in the typical repertoire of Western classical music. The compilation of a representative dataset constitutes a cumbersome task since collecting and annotating data is time-consuming and judgement of “importance” or “appropriateness” of works is highly subjective. In our work, we focused on composers whose works frequently appear in concerts and on classical radio programs. At the same time, we tried to ensure a certain variety of countries, composers, musical forms, keys, or tempi.

For classification experiments, a balanced distribution of instances with respect to the class labels is beneficial. For these reasons, we compiled a dataset of $4 \times 400 = 1600$ pieces,¹ which we assigned to the four historical periods Baroque, Classical, Romantic, and Modern² (Table 7.1). Our manual attribution of pieces to these coarse-level *periods* or *eras*³ is rather subjective and not unambiguous. We tried to focus on such composers where we expect musicologists to agree about the era assignment and checked this assumption with categorization in Wikipedia.⁴ Later, we will discuss our selection guidelines in more detail. In the following, we refer to this corpus as *Cross-Era* dataset. We used this data for the classification experiments in Chapter 8 and in the associated publications [256, 258, 259].

To systematically investigate the timbre invariance of our algorithms, we further balanced the dataset with respect to the instrumentation. For every period, the dataset incorporates each 200 pieces of orchestra and piano music. To avoid the system learning timbral particularities (when classifying on piano only), we only selected piano recordings performed on

¹For multi-movement works or work cycles, we regard every movement as a “piece” when counting items in the dataset. Moreover, global feature values are also computed on the movement level.

²Hereby, the “Modern” class mainly refers to works from the first half of the 20th century. We did not include works that are stylistically close to late Romanticism.

³In this thesis, we synonymously use the terms *period* and *era*.

⁴<http://www.wikipedia.org>

Table 7.1. Cross-Era dataset. For the four eras under consideration as well as for the “Add-On” data, we list the composers and their countries for each sub-class.

<i>Era</i>	<i>Instrument.</i>	<i>Composers</i>	<i>Countries</i>
Baroque	Piano	Bach, J. S.; Couperin, F.; Giustini, L.; Platti, G. B.; Rameau, J.-P.	France, Germany, Italy
	Orchestra	Albinoni, T.; Bach, J. S.; Corelli, A.; Handel, G. F.; Lully, J.-B.; Purcell, H.; Rameau, J.-P.; Vivaldi, A.	England, France, Germany, Italy
Classical	Piano	Cimarosa, D.; Clementi, M.; Dussek, J. L.; Haydn, J.; Mozart, W. A.	Austria, Czechia, England, Italy
	Orchestra	Bach, J. C.; Boccherini, L. R.; Haydn, J. M.; Haydn, J.; Mozart, W. A.; Pleyel, I. J.; Salieri, A.	Austria, England, Germany, Italy
Romantic	Piano	Brahms, J.; Chopin, F.; Faure, G.; Grieg, E.; Liszt, F.; Mendelssohn Bartholdy, F.; Schumann, C.; Schumann, R.; Tchaikovsky, P. I.	France, Germany, Hungary, Norway, Poland, Russia
	Orchestra	Berlioz, H.; Borodin, A.; Brahms, J.; Bruckner, A.; Dvořák, A.; Grieg, E.; Liszt, F.; Mendelssohn Bartholdy, F.; Mussorgsky, M.; Rimsky-Korsakov, N.; Saint-Saëns, C.; Schumann, R.; Smetana, B.; Tchaikovsky, P. I.; Verdi, G.; Wagner, R.	Austria, Czechia, France, Germany, Hungary, Italy, Norway, Russia, USA
Modern	Piano	Bartók, B.; Berg, A.; Boulez, P.; Hindemith, P.; Messiaen, O.; Milhaud, D.; Prokofiev, S.; Schönberg, A.; Shostakovich, D.; Stravinsky, I.; Webern, A.	Austria, France, Germany, Russia, USA
	Orchestra	Antheil, G.; Bartók, B.; Berg, A.; Britten, B.; Hindemith, P.; Ives, C. E.; Messiaen, O.; Prokofiev, S.; Schönberg, A.; Shostakovich, D.; Stravinsky, I.; Varèse, E.; Webern, A.; Weill, K.	Austria, England, France, Germany, Hungary, Russia, USA
“Add-On”	Piano	Bach, C. P. E.; Beethoven, L. van; Debussy, C.; Ravel, M.; Scarlatti, D.; Schubert, F.; Sibelius, J.; Weber, C. M. von	Austria, France, Finland, Germany, Italy
	Orchestra	Bach, C. P. E.; Beethoven, L. van; Debussy, C.; Mahler, G.; Mozart, Leopold; Ravel, M.; Rossini, G., Scarlatti, D.; Schubert, F.; Sibelius, J.; Stamitz, Johann; Strauss, R.; Telemann, G. P.; Weber, C. M. von	Austria, Czechia, France, Finland, Germany, Italy

the modern grand piano (no harpsichord recordings in the Baroque class). Moreover, the orchestral data neither includes works featuring vocal parts nor solo concertos.⁵ For obtaining a meaningful subgenre classification rather than capturing individual composer styles, every category contains music from a minimum of five different composers from three different countries. Table 7.1 lists the composers and the countries for each sub-class.

To make sure that we do not classify properties other than style-related ones, we tried to include a certain range of different works by every composer. Hereby, we considered different musical forms (sonatas, variations, suites, symphonies, symphonic poems, overtures, and more) as well as fast and slow movement types (head movements, slow movements, minuets, etc.). The data exhibits a variety of keys and modes (major/minor) but is not perfectly balanced with respect to these aspects.

⁵Because of the omnipresence of the figured bass, it is hard to find recordings for Baroque orchestral works without involving a harpsichord. This may lead to some timbral peculiarity for the Baroque orchestra class. Nevertheless, the harpsichord may not be too present acoustically since it constitutes an accompanying instrument in these pieces.

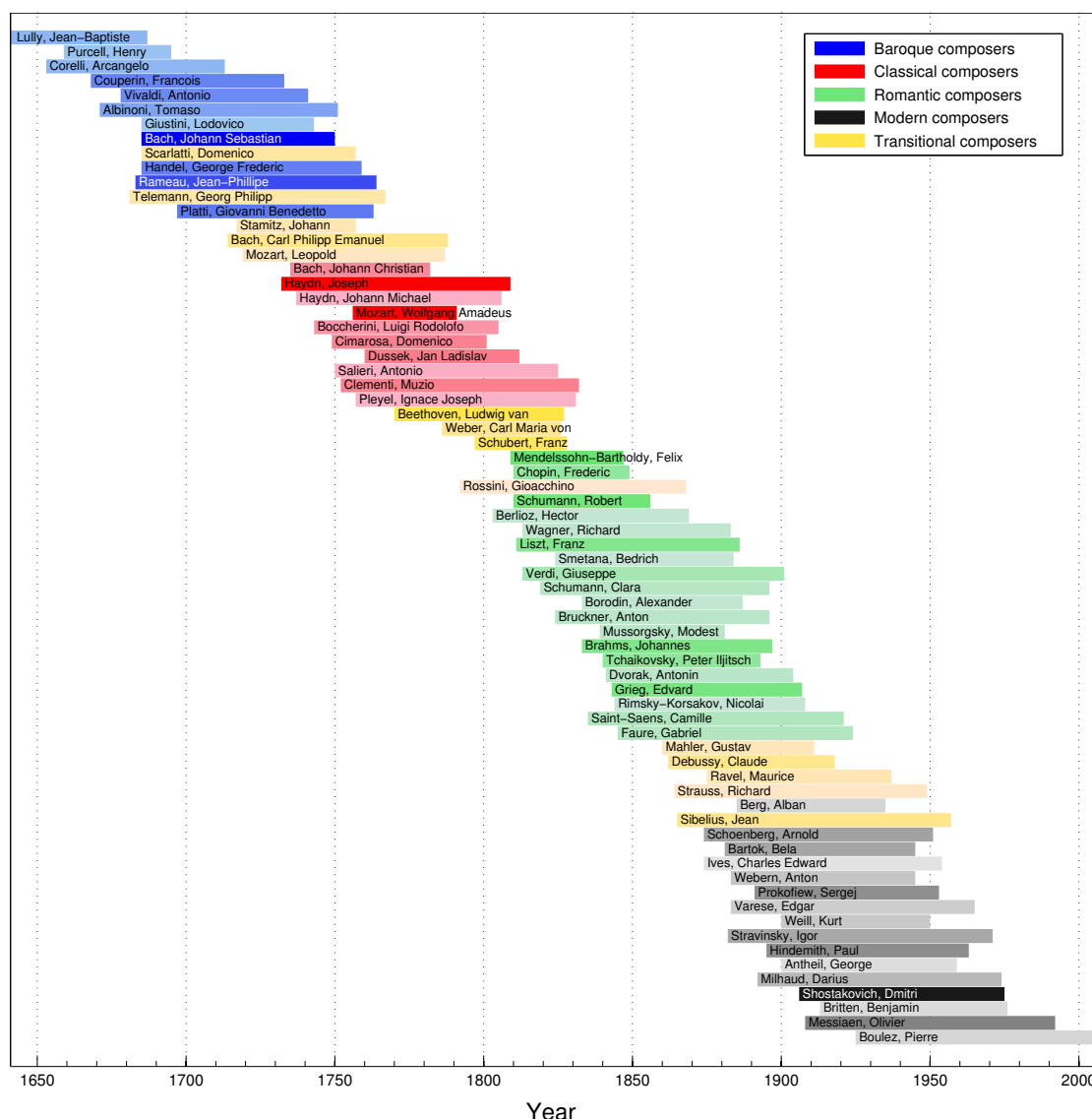


Figure 7.1. Overview of the composers in the combined dataset. A bar corresponds to the composer’s lifetime. The color marks the class a composer belongs to. Yellow bars refer to the “Add-On” data. With the intensity of the color, we indicate the number of the composer’s works considered in the dataset. More intense colors correspond to a higher number (see, for example, J. S. Bach, or W. A. Mozart).

From a musicological point of view, a categorization into four eras is rather superficial. In the classification experiments (Chapter 8), however, we want to test our features’ capability for a very rough style analysis. We therefore try to avoid ambiguous musical tasks that treat subtle stylistic differences. To this end, we did not include composers whose stylistic attribution is rather ambiguous.⁶ As a consequence, the *Cross-Era* data does not show an equal distribution with respect to the composers’ lifetimes but exhibits some historical “gaps.” To overcome this problem, we created an additional set of recordings comprising works by such “transitional” composers. This “Add-On” includes each 200 piano and orchestra pieces and serves to “fill the gaps” between the historical periods in the *Cross-Era* set. The

⁶For example, we did not select works by Beethoven or Schubert since these composers show influences from both Classical and Romantic styles.

transitional character mainly relates to the composers' lifetime (e. g., for Carl Maria von Weber or Franz Schubert). Some of the composers contributed to the establishment of a new style—such as Johann Stamitz or Carl Philipp Emmanuel Bach in the pre-classical phase. In other cases, we included composers who historically but not stylistically match one of the eras. Examples are Richard Strauss and Jean Sibelius whose style could be considered closely connected to the Romantic era rather than to 20th century's avant-garde (which we mainly consider for the Modern period). We end up with a more or less balanced distribution (Figure 7.3), which enables us to analyze the correlation of style characteristics with composition time in this section.

The lower part of Table 7.1 lists the additional composers. Figure 7.1 provides a visualization of the combined dataset with respect to the composers' lifetime. The colors mark the class labels with the yellow bars corresponding to the “transitional” composers. With the intensity of the color, we indicate the number of recordings included in the dataset by the respective composer. Popular composers such as Johann Sebastian Bach, Wolfgang Amadeus Mozart, or Dmitri Shostakovich contribute more works than others. Following this principle, our dataset may—to some degree—represent the typical repertoire of Western classical music. We refer to the combined dataset as *Cross-Era+Add-On* comprising 2000 tracks in total.

We compiled the recordings from commercial audio CDs. In order to allow reproduction of some of our experiments, we published the basic audio features on a website.⁷ We provide chroma features (Section 3.5.2) and chord analysis results, which served as basis for the experiments presented in this chapter.

7.2 Visualization of Audio Features through Music History

7.2.1 Data Mapping

To examine the stylistic evolution of music over the history, a corpus of works with a roughly equal distribution of composition dates would be necessary. Unfortunately, we do not have these composition dates for all pieces in our dataset. A huge effort would have to be made to compile all this information—and for many works, the composition years are unknown or in doubt. Even if we had all composition dates at hand, it would still constitute a difficult task to find an equal amount of works for all the years while—at the same time—balancing the dataset with respect to other aspects such as the instrumentation.

For these reasons, we use a different strategy and map the works of a composer onto his or her lifetime. Figure 7.2 illustrates this procedure in detail. This approach is rather superficial since, with this simplification, we cannot resolve historical details of style evolution. In particular, the assumption of stylistic homogeneity over a composer's lifetime may be violated in some cases. We may think of composers with several “creative periods” such as Arnold Schönberg whose style developed from late Romanticism to dodecaphony in several steps. In this chapter, however, we are interested in a rather “global” view and look at the overall tendencies. For this reason, we assume that the simplifications of our mapping technique do not have a crucial impact for analyzing the general trends.

With the above mentioned procedure, our dataset spreads over the historical timeline as shown in Figure 7.3. Though not being a flat distribution, we have at least five compositions on average for every year from 1700 to 1950. Before 1700 and after 1950, the average number

⁷<http://www.audiolabs-erlangen.de/resources/MIR/cross-era>

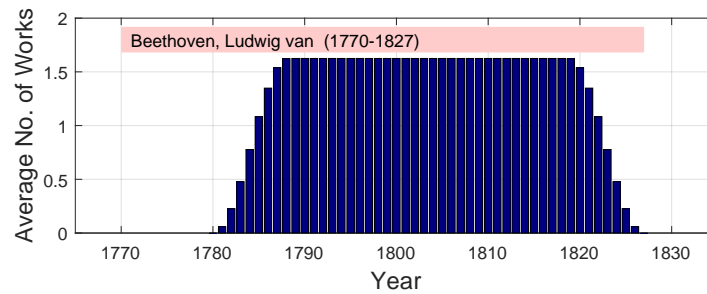


Figure 7.2. Example distribution of a composers works over the lifetime. Here, we show the process of mapping a composer’s works onto the lifetime for L. van Beethoven, living 1770–1827 and contributing 63 movements to the dataset. For this, we use a Tukey window with parameter $\alpha = .35$ while excluding the first ten years of the composer’s lifetime. We normalize the years’ values so that their sum equals the total number of the composers’ works in the dataset.

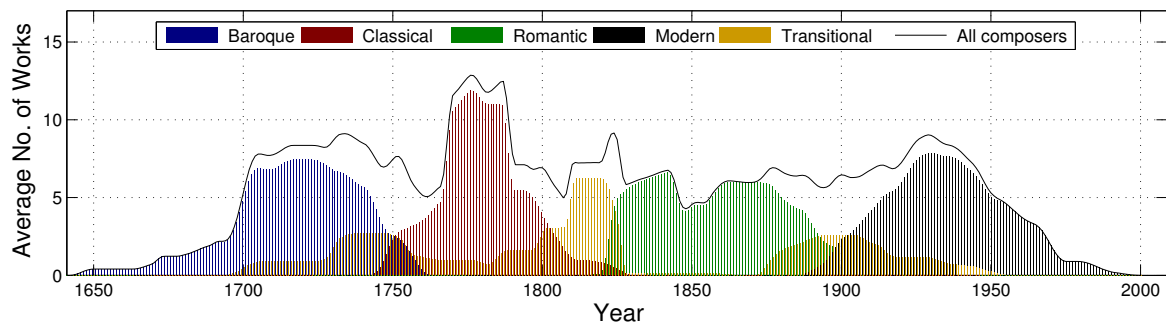


Figure 7.3. Average number of works per year for the different eras. The colors indicate the eras Baroque, Classical, Romantic, and Modern. The yellow bars correspond to the additional works by transitional composers between the eras. The black line denotes the total number of works per year in the dataset.

of pieces—and composers—decreases. For this reason, we need to be very careful with an interpretation of the results for these outer time spans since they may be heavily biased towards the pieces of only one or two composers. In subsequent sections, we use this mapping procedure to visualize values of features over the time axis. For this, we first compute the feature values for all pieces of a composer and average. Then, we map the average features to years using the respective weighting factors. As for normalization, we finally divide the year-wise values by the number of works in the year—given by the black line in Figure 7.3—so that a constant feature value for all pieces results in a flat curve.

7.2.2 Analysis of Chord Progressions

In Section 2.6.3, we introduced the categorization of chord progressions into authentic and plagal types as proposed by Bárdos [14]. According to [69], the quantitative relation between authentic and plagal progressions provides a useful criterion to discriminate musical styles.

Motivated by such hypotheses, we now want to use our mapping technique for analyzing chord progressions over the course of music history. For estimating the chords, we use the public algorithm Chordino.⁸ This method relies on NNLS chroma features (see Section 3.5.3) and incorporates Hidden Markov Models for concurrently estimating and smoothing the chord labels [147]. With the public software Sonic Annotator, we extracted the chords for our database.

⁸<http://isophonics.net/nnls-chroma>

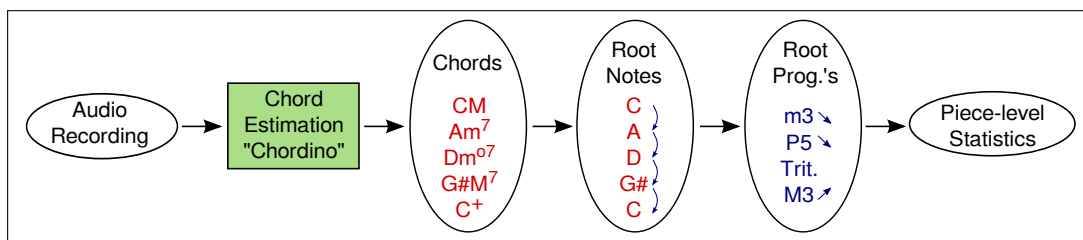


Figure 7.4. Estimation of root note progressions. In this schematic overview, we show the processing flow for estimating the frequency of root progressions. First, we reduce the output of the chord estimator to consider only root notes. From this sequence, we calculate statistics of melodic intervals between the root notes.

The Chordino plugin allows for an adaptation of possible chord types using a dictionary file (“chord.dict”). We modified this dictionary for our purpose by only using the four basic triad types (see Figure 2.11) as well as the five seventh chord types presented in Figure 2.13. In the appendix, we show the dictionary file in detail (Table A.1). We do not use the bass note estimation since, for classical music, the bass notes⁹ do not necessarily lie within a fixed pitch range. For all other system parameters, we use the default values. Of course, this automatic chord estimation system produces a number of errors or results that are not musically meaningful or accurate. Furthermore, the chosen selection of chord types may not be suitable for the sonorities appearing in the Modern class, in particular. This means that, for the Modern pieces, a specific type of “measurement error” may be characteristic rather than an explicit output that is semantically meaningful. Nevertheless, we expect certain tendencies to occur since we look at a large number of works and, thus, local errors may disappear in the global view. Moreover, errors concerning the chord *types* do not affect some of our experiments since we are mainly interested in the chords’ *root notes* and their progressions.

As a first scenario, we only consider such root note progressions. To this end, we only keep the root notes of the chords and count the melodic intervals between them (see Figure 7.4). We divide the resulting numbers by the total number of chord progressions to obtain relative values for each piece. With the method presented in Section 7.2.1, we then map these piece-level features onto the time axis (Figure 7.5). We arrange the values according to authentic (falling) and plagal (ascending) progressions following the system by Bárdos. For details on this theory, we refer to Section 2.6.3 and Table 2.2. Because of enharmonic equivalence in our features, we cannot assign the tritone progressions to one of these categories (CM → F#M equals CM → GbM). We do not consider transitions between chords with the same root note either such as, for instance, the transition CM → Cm.

We now apply this analysis to the whole *Cross-Era+Add-On* dataset using our mapping technique (Figure 7.5). Here, we first observe the important role of the fifth progressions. Both authentic and plagal fifth progressions occur frequently, with a slight dominance of the authentic fifth—especially for the early 18th century. Another important step is the major second. Here, both directions show similar rates. During the 19th century, third note progressions seem to become more important. From the year 1900 on, the distribution flattens slowly. Moreover, the number of minor second and tritone progressions increases. Overall, the flat distribution may point to a random-like behavior of the chord assignment. This is in accordance with our expectation, since the chord types allowed for the system

⁹Here, we refer to the *harmonic* bass note—the lowest note in a given voicing of a chord—independently of this note’s octave or the playing instrument.

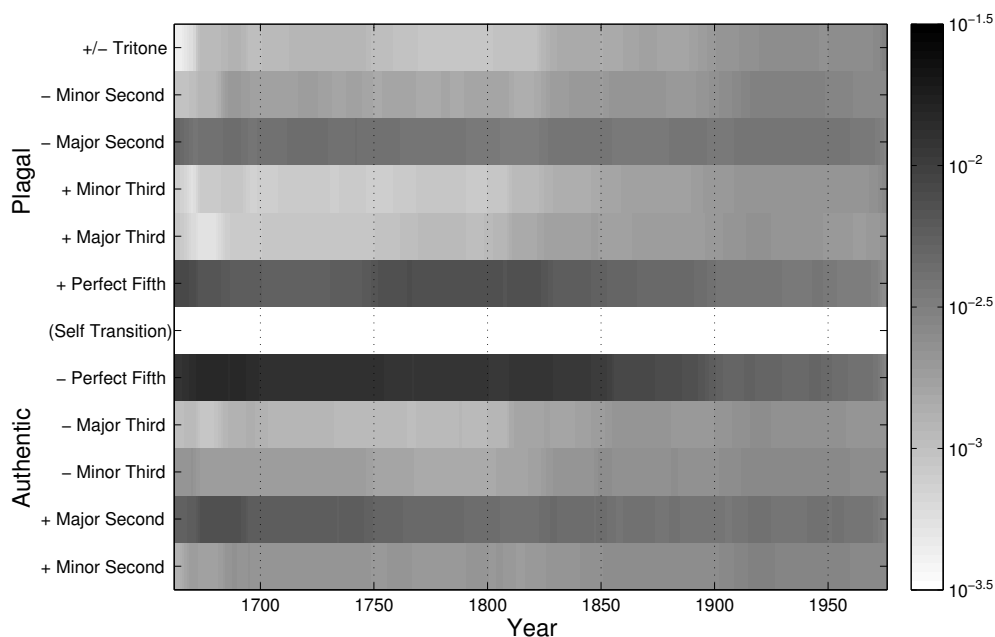


Figure 7.5. Relative frequency of root note progressions. With the mapping method from Section 7.2.1, we visualize the frequency of melodic intervals between root notes. We arrange the progressions according to authentic and plagal categories. Hereby, we ignore self-transitions (between chords with identical root notes). The gray levels (logarithmic color-axis) indicate the relative frequency of the root note distances.

are not relevant for most of the Modern class pieces. To see the influence of the individual composers' pieces on the chord progressions distribution, we show in the appendix a detailed plot with composer-specific root progressions (Figure A.1).

To systematically evaluate the relation between authentic and plagal progression, we sum up all progressions belonging to each group (see Table 2.2). Here, we ignore the tritone- and self-transitions. For each piece, we calculate the ratio between the piece-wise normalized numbers $\#Authentic/\#Plagal$. A ratio of 1 indicates an equal numbers of plagal and authentic progressions. We map these numbers onto the time axis with the procedure presented in Section 7.2.1. Figure 7.6 shows the resulting curve. With a bootstrapping procedure, we estimate the robustness of the year-wise mean. This method serves to analyze the stability of the mean when the underlying distribution is unknown. For each year, we create 500 duplicates of the initial sample (the feature values contributing to this year) using sampling with replacement.¹⁰ We calculate the mean from each of the 500 samples and derive the 95 % confidence interval. This bootstrap error is larger for years with only few contributing pieces such as the years before 1700.

Looking at Figure 7.6, we always find a higher number of authentic progressions (ratio > 1). This points to a high importance of progressions such as authentic cadences or “circle of fifths” sequences, which are typical for a “functional” concept of harmony. Around the year 1750, we find a considerable decrease of the ratio. Looking at the composer plot (Figure 7.1), several typical Baroque composers stop contributing here (J. S. Bach, G. F. Handel, J. P. Rameau, and others). For this reason, the dominance of authentic progressions may be a criterion to discriminate late Baroque from Classical style. Between the years 1820–1850, we find a small

¹⁰Sampling with replacement leads to a sample of the same size but usually with some values missing and others occurring multiple times. The weights for the individual composers (Figure 7.2) serve as sampling probabilities.

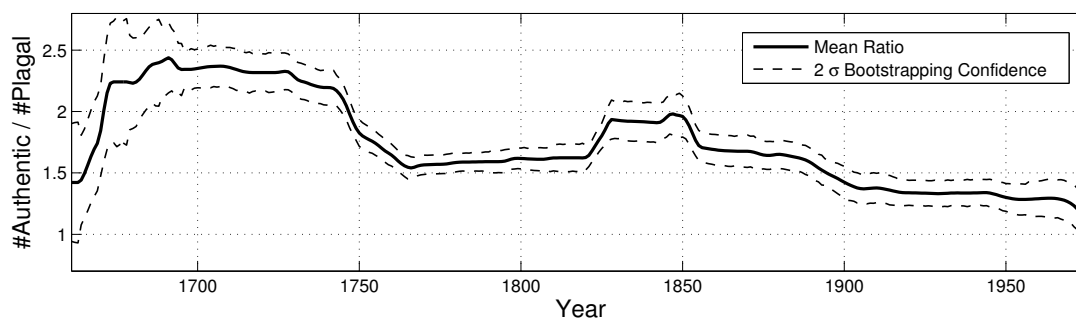


Figure 7.6. Ratio between authentic and plagal chord progressions distributed over the years. For each year, we performed weighted bootstrapping (500 bootstrap samples) on the piece-wise values and calculated a 2σ confidence interval (95% confidence).

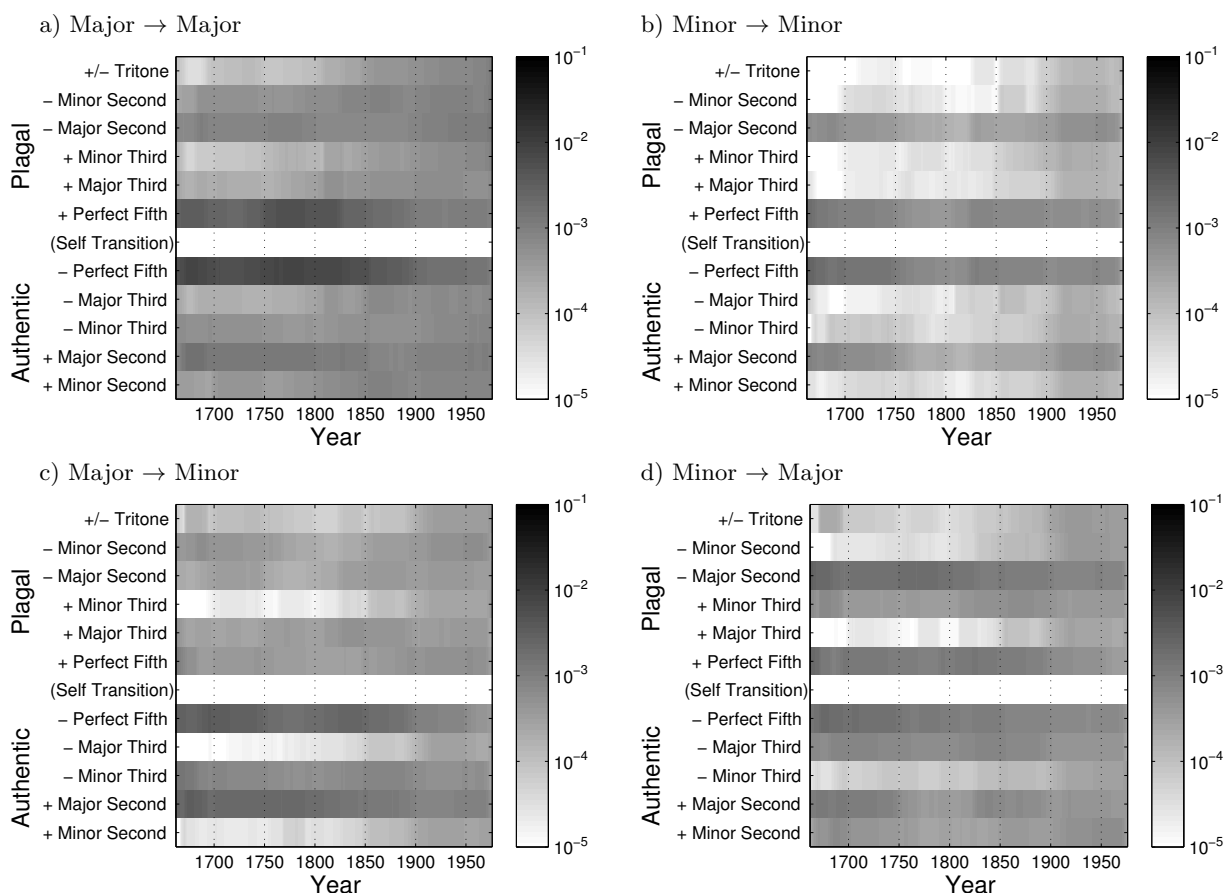


Figure 7.7. Root note progressions for different chord types. Here, we order the bigrams by the quality of the first and last chord. We order the root progressions to authentic and plagal categories.

increase of authentic progressions. Among others, we find contributions by R. Schumann and F. Mendelssohn Bartholdy here. Possibly, a new popularity of the Baroque music in this time showed some influence on the style of these composers.¹¹ Besides such speculations, the reasons for this behavior are not clear and have to be examined in future work. During the 20th century, the ratio gradually comes closer to 1. This confirms our expectation of

¹¹For example, many treatises on music history consider the rediscovery and performance of J. S. Bach's "St. Matthew Passion" initiated and conducted by F. Mendelssohn Bartholdy in 1829 as an important event.

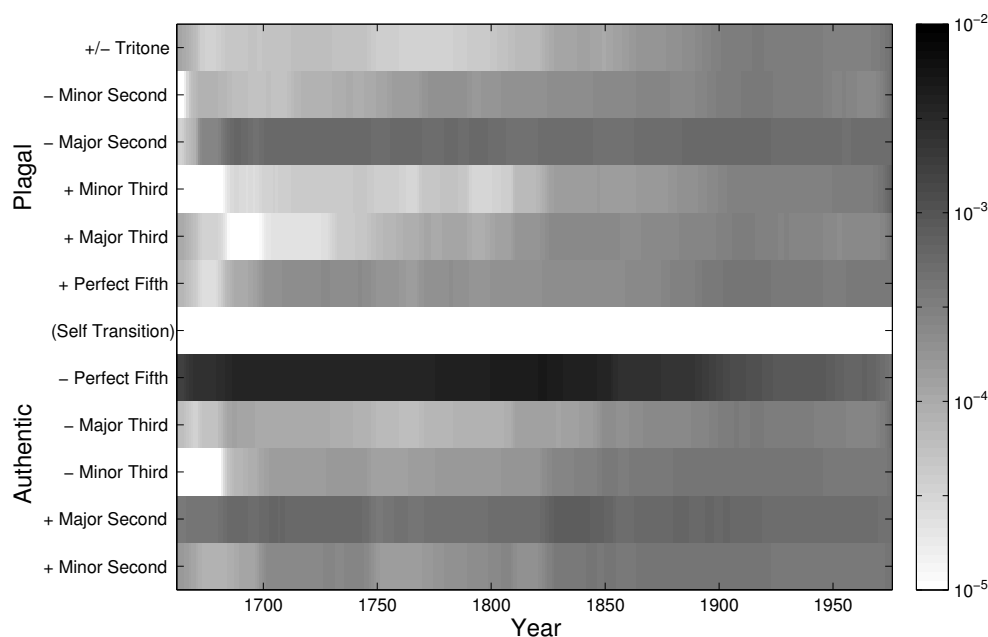


Figure 7.8. Root note progressions of a dominant seventh chord. For this plot, we sum up all progressions from a dominant seventh chord to any other chord type ($M^7 \rightarrow M, M^7 \rightarrow m, \dots, M^7 \rightarrow m^7, \dots$). We see the prevalence of the authentic fifth progression in this scenario.

a random-like chord estimation, which should not exhibit a trend towards certain types of chord progressions.

To study the influence of the chord types, we want to discriminate between major and minor types. “Major type” refers to all chords based on the major triad and, thus, includes the major triad itself (M), the major seventh chord ($M^{\text{maj}7}$), and the dominant seventh chord (M^7). The minor type comprises the minor triad (m) and the minor seventh chord (m^7). In Figure 7.7, we show the progressions by type. As an example, “Major \rightarrow Minor” refers to all bigrams beginning with a major type chord and ending with a minor type chord—arranged according to the distance of the chords’ root notes. For all combinations, we find a rough similarity to Figure 7.5. Root progressions by perfect fifth and major second intervals seem to be important for all combinations. When the first chord is of major type, the authentic progressions seem to be more frequent (Subfigures a) and c)).

This behavior becomes more evident when we only look at progressions departing from a dominant seventh chord (M^7) and leading to a chord of any other type (Figure 7.8). In this case, the authentic fifth progressions is much more frequent than any other resolution (up to factor 10). This is no surprise since, in common-practice music, the dominant seventh chord typically resolves in that way—such as for the frequent cadences $V^7\text{-I}$ and $V^7\text{-i}$.

Finally, we want to show the distribution of recognized chord types over the years (Figure 7.9). Here, we find a dominance of “stable chords” with a major or minor triad as basis. During the Classical period (about 1750–1820), the major chord types are even more present compared to other types. The diminished types gain importance during the 19th century. The augmented type—which only comprises the augmented triad here—is found more often in 20th century pieces. Looking at the seventh chords, we also see a decreasing influence during the Classical time. The diminished seventh chord ($^{\circ}7$) seems to be particularly important during the 19th century. In contrast, the half-diminished seventh chord ($^{\flat}7$) becomes more important with the end of the 19th century.

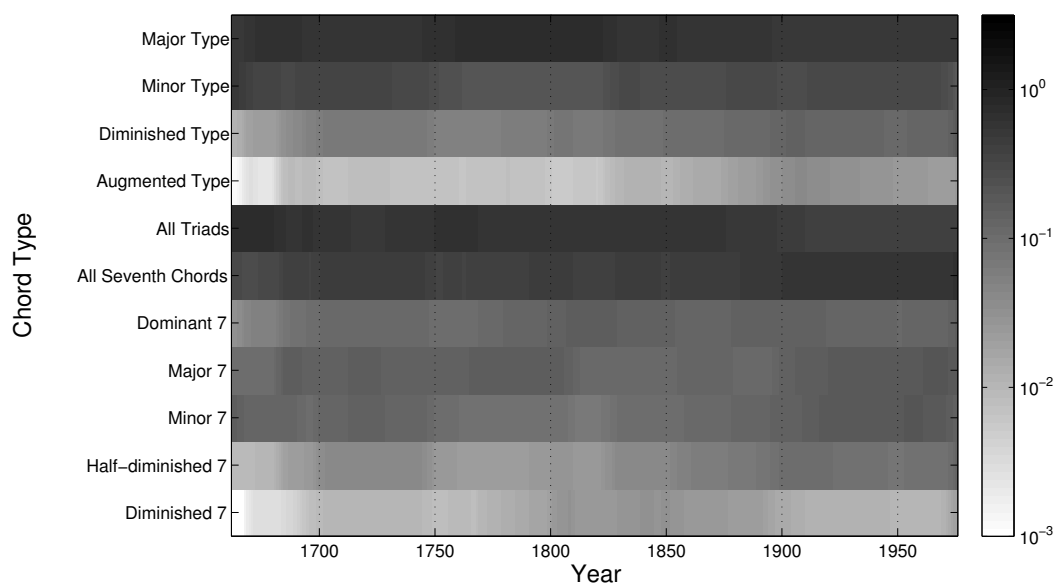


Figure 7.9. Chord types distributed over the years. “Major Type” includes all major triads as well as all seventh chord types based on a major triad. “Minor type” are the minor triad and the minor seventh chord.

In Figure 7.9, we detect some problems with the detailed chord type analysis. In this plot, the major seventh chord is sometimes even more frequent than the dominant seventh chord—for example, from 1750–1800. However, this chord was practically not existing at that time. In our interpretation, this is mostly a misinterpretation of the major triad by the Chordino algorithm. This may result from the third partial of the triad’s third note, which corresponds to the major seventh above the root. Another reason may be the presence of the seventh as a figurative melodic note. Because of such effects, we have to be very careful with a comparison of chord types. Nevertheless, most of the confusions do not lead to a wrong *root note* estimation and, thus, produce no errors when analyzing root note progressions.

7.2.3 Analysis of Interval and Complexity Features

In Chapter 6, we presented several features for quantifying the presence of interval classes or tonal complexity. In contrast to the chord estimation used for the previous section, these features do not have to locally decide on the best matching item. They have a continuous-valued output and, thus, can reflect mixtures of items. In this section, we want to analyze the distribution of such features over the course of music history.

First, we use a set of features ($\Psi^{IC1}, \dots, \Psi^{IC6}$) describing interval categories as presented in Section 6.1.3. We calculate the features on the basis of NNLS chromagrams with a resolution of 10 Hz (no feature smoothing). For this reason, the features mainly refer to simultaneous intervals. Then, we map the results onto composition years using the strategy from Section 7.2.1. Figure 7.10 shows the resulting plot. We observe a prominent role of the category IC5 comprising perfect fifth and fourth intervals. During the 20th century, the frequency of these intervals slightly decreases and the overall distribution flattens. We saw a similar behavior for the chord progressions in the previous section. The major third and minor sixth class (IC4) seems to be important for the Classical and Romantic periods. For the minor third, we find an increase during the “high Romanticism” (about 1830–1890). One reason may be the frequent use of chords such as the diminished seventh chord (^{o7}) in this

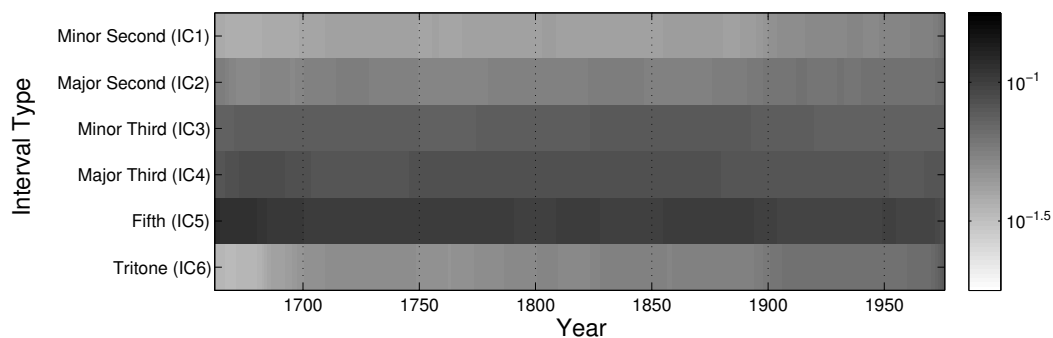


Figure 7.10. Interval type features distributed over the years. Here, we show the template-based features from Section 6.1 for quantifying interval types over the time axis. The gray levels indicate the average feature values for each year (logarithmic color axis). Note that the interval inversion (complementary intervals) cannot be resolved. For example, “Minor Third” also describes a major sixth.

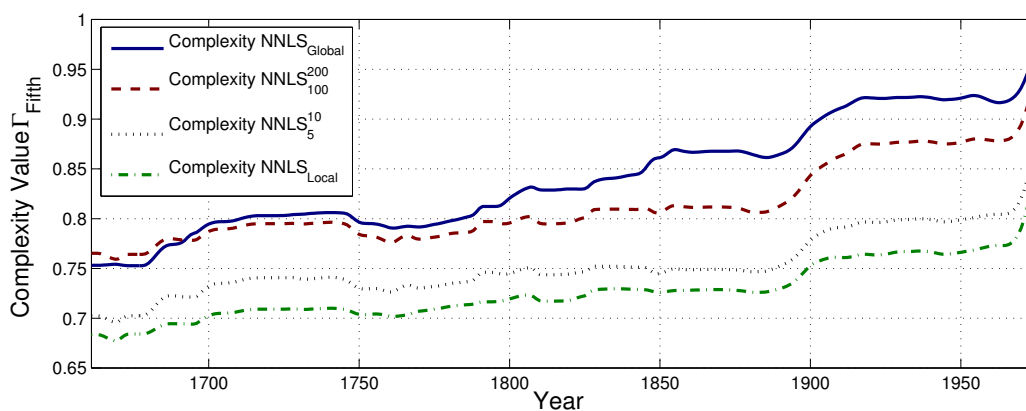


Figure 7.11. Complexity features distributed over the years. This feature Γ_{Fifth} (Equation (6.24)) describes the pitch distribution over a circle of perfect fifths (see Section 6.2.3). We compute the features for different resolutions of the NNLS chroma features and plot the average feature values per year.

period—which we observed in Figure 7.9. In the 20th century, the dissonant categories IC2 (whole tone) and, in particular, IC1 (semitone) and IC6 (tritone) become important. We expected such behavior since 20th century composers often use dissonant chords such as, for example, chromatic clusters. Fucks and Lauter [66] presented similar results when statistically analyzing melodic and harmonic intervals in single parts (violin, flute, and vocal parts) based on symbolic data. In particular, they observed a prominent role of the intervals M7 and m9—both belonging to IC1—in works by A. Schönberg and A. Webern.

Second, we visualize measures for quantifying tonal complexity over the years. In Section 6.2, we proposed such measures and analyzed their behavior for single chords or segments of pieces. Here, we calculate the feature Γ_{Fifth} (Equation (6.24)) for four different time scales on the basis of NNLS chroma features ($\text{NNLS}_{\text{global}}$, NNLS_{100}^{200} , NNLS_5^{10} , $\text{NNLS}_{\text{local}}$).¹² We average the features over each piece and distribute the values over the history as presented above (Figure 7.11). For all temporal resolutions, we find a general increase with the years. After 1750, the complexity seems to decrease for some decades. Interestingly, this confirms the demand for more “simplicity”, which musicologists often claim to be a paradigm for the begin of the Classical period. We observe a similar behavior—but less obvious—for the early Romantic period (about 1810–1830). After this time, the *global* complexity considerably in-

¹²For the details of the chroma smoothing procedure, we refer to Section 3.5.5.

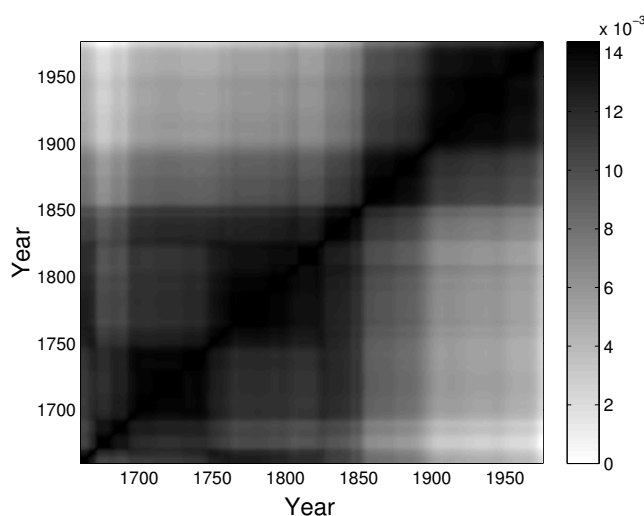


Figure 7.12. Self-similarity matrix of root note progressions. This matrix shows the Euclidean distance between each pairs of years encoded by the gray levels. As input features, we used the eleven basic root note progressions between any types of chord and mapped them onto the years.

creases during the 19th century, whereas the *local* complexity stays approximately constant. In our interpretation, this effect may stem from an increasing use of modulations—flattening the global chroma histogram—whereas the local structures such as chords remain less complex. This relationship changes towards the 20th century, where we observe a strong increase of complexity for all temporal scales. This means, we also find complex local sonorities for the 20th century, which may arise from contributions of rigorous atonal music by composers such as A. Schönberg, A. Webern, and others. As mentioned above, we have to be careful with the early and late years shown in our plots. In particular, the sharp increase at around 1970 may not be representative for this time. This artifact is caused by the pieces of P. Boulez and his teacher O. Messiaen, which are the only composers contributing to these years. For studying the composer-specific complexity values, we show a detailed plot in the appendix (Figure A.2).

7.3 Style Analysis with Clustering Methods

7.3.1 Clustering Years

7.3.1.1 Chord Progressions

In the previous section, we presented a method for mapping feature values of individual pieces onto a time axis. We applied this technique for analyzing automatically extracted chord progressions as well as interval and complexity features over history. At first glance, some of the observed structures relate to stylistic evolutions in music history. We now want to apply unsupervised clustering techniques to analyze the similarity of pieces, composers, and years on the basis of our features. This may provide an insight in the usefulness of such features for stylistic analysis.

We first analyze the chord progression statistics individually. For this, we look at the root note progression statistics mapped onto the years as presented in Figure 7.5. We consider the years 1761–1975 where at least three composers contribute to the statistics. Since we ignore the self-transitions, we end up with eleven progressions and, thus, a feature matrix

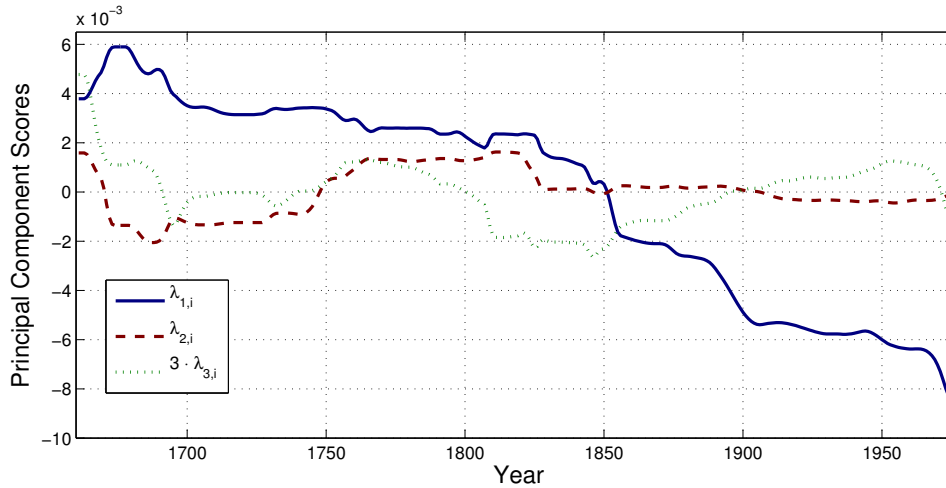


Figure 7.13. First three principal components from eleven root progression types. We display the PCA scores over the instances (years). To better recognize the small component $\lambda_{3,i}$, we multiplied its value with the factor 3.

$\mathcal{F}_{\text{RootProg}} \in \mathbb{R}^{11 \times 315}$. On this data, we calculate a self-similarity matrix (Figure 7.12). This matrix comprises the Euclidean distance between each possible pair of data points. Blocks of higher values indicate a higher homogeneity of the respective fragments. We can observe several of such blocks. Two pairs of blocks—with a separation at about 1850 and 1900—suggest a discrimination into the categories traditional–modern (or tonal–atonal). Furthermore, we find darker blocks for the years 1700–1750 (late Baroque) and 1750–1820 (Classical period). In contrast, the 19th century does not constitute a homogeneous period in this plot.

To analyze the contributions of the individual progressions, we perform principal component analysis (Section 3.6.4.1). For normalization, we first subtract from each row its mean value.¹³ Then, we compute the principal component weights $\mathbf{w}^l \in \mathbb{R}^{11}$ and scores $\lambda_{i,l} \in \mathbb{R}$ with $l \in [1 : 11]$ and $i \in [1 : 315]$. In Figure 7.13, we show the PCA scores. The scores constitute the feature values (linear combinations of the root progressions for each year) in the principal component space¹⁴ of dimension $\mathbb{R}^{11 \times 315}$. Table 7.2 lists the weights \mathbf{w}^l for the first three components $l \in [1 : 3]$. The weight vectors are normalized to $\ell_2(\mathbf{w}^l) := 1$. A minus sign indicates negative contribution.

The first component score decreases over time and seems to capture the difference between the early periods and the rather modern styles. Looking at the weight vector \mathbf{w}^1 in Table 7.2, we see the largest entries for the perfect fifths progressions with an emphasis on the authentic P5 (.871). Only the perfect fifth and major second progressions have positive sign, in contrast to all other components. Thus, the first component describes the relative frequency of the most typical progressions (perfect fifths and major seconds) in tonal music. From 1850 on, other progressions seem to become more frequent leading to a smaller value of the first principal component.

¹³For features of different type, a division of each row’s values by the standard deviation is also necessary. Here, we have features of similar type. We do not divide by the standard deviation in order to keep the influence of the overall frequency of a chord progression type.

¹⁴In Section 3.6.4.1, we introduced PCA as a method for dimensionality reduction. In general, the principal component space has the same size as the initial feature space. To obtain a reduced number of dimensions, we usually keep only a fraction of the principal components. This is useful since, with increasing index l , a vector \mathbf{w}^l describes a smaller fraction of the data’s variance.

Table 7.2. Principal component weights for root note progressions. We re-ordered the vector entries according to the axis of Figure 7.5. The second column Δ indicates the size of the respective interval in semitones. Note that we cannot resolve the direction and therefore, the values may also refer to the complementary interval in opposite direction ($P5 \searrow \cong P4 \nearrow$).

Interval	Δ	w^1	w^2	w^3	Quality
+4 \nearrow	+6	-.138	-.178	-.045	None
m2 \searrow	-1	-.127	-.159	-.012	Plagal
M2 \searrow	-2	.038	-.155	.358	Plagal
m3 \nearrow	+3	-.139	-.039	-.136	Plagal
M3 \nearrow	+4	-.121	.068	-.330	Plagal
P5 \nearrow	+7	.325	.715	.407	Plagal
P5 \searrow	-7	.871	-.202	-.418	Authentic
M3 \searrow	-4	-.114	-.039	-.250	Authentic
m3 \searrow	-3	-.081	-.125	-.021	Authentic
M2 \nearrow	+2	.199	-.579	.576	Authentic
m2 \nearrow	+1	-.082	-.095	-.087	Authentic

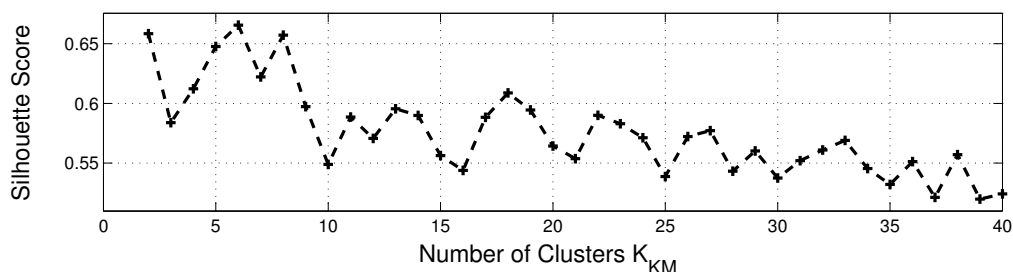


Figure 7.14. K-means clustering for root note progressions. For each value of K_{KM} , we repeat the clustering 200 times. We show the mean silhouette score over all runs indicating the clustering quality.

The second component’s weight vector w^2 also has large values for the perfect fifth progressions—but with opposite sign. The plagal P5 has a large positive coefficient (.715), whereas all authentic progressions (including P5 and M2) have negative coefficients. This means that the second component describes some kind of *ratio between plagal and authentic progressions*. Looking at the corresponding PCA score in Figure 7.13, we see that this component mainly distinguishes the Classical period (about 1750–1820) from the other years. In our opinion, this observation is interesting since it stems from an unsupervised clustering of the progression features—without any prior assumptions about style periods.

To obtain an automatic partitioning of the years into segments, we run the K -means clustering algorithm (Section 3.6.2.1) on the three principal components of our chord progression features. For this method, the number of clusters K_{KM} is an important parameter. To determine the optimal value, we calculate so-called “silhouette scores” for $K_{KM} \in [1 : 40]$. The silhouette is computed for every data point (year) and indicates how similar that point is to points in its own cluster compared to points in other clusters [207]. A high silhouette score indicates a good clustering. Figure 7.14 shows the scores over K_{KM} . Six or eight clusters seem to be optimal for this data. However, we also obtain a high score for two clusters.

In Figure 7.15, we plot the results of the clustering procedure. Interestingly, the first split point (for two clusters) divides the romantic period at about 1850. We find several “stable” cluster boundaries that are present in the clustering results for most K_{KM} values. Here, they arise at about 1750, 1850, and 1900. For all K_{KM} , the 19th century is split into several

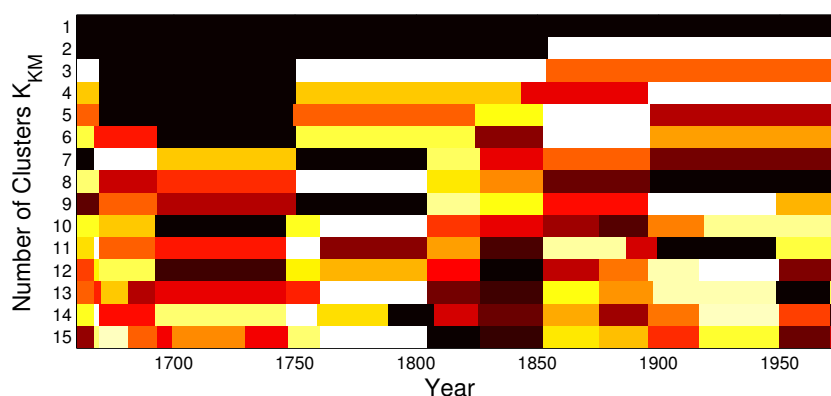


Figure 7.15. Clustering of years for root note progressions. For different numbers of clusters K_{KM} , we show the clustering result based on root note progressions. Each cluster is indicated by a color.

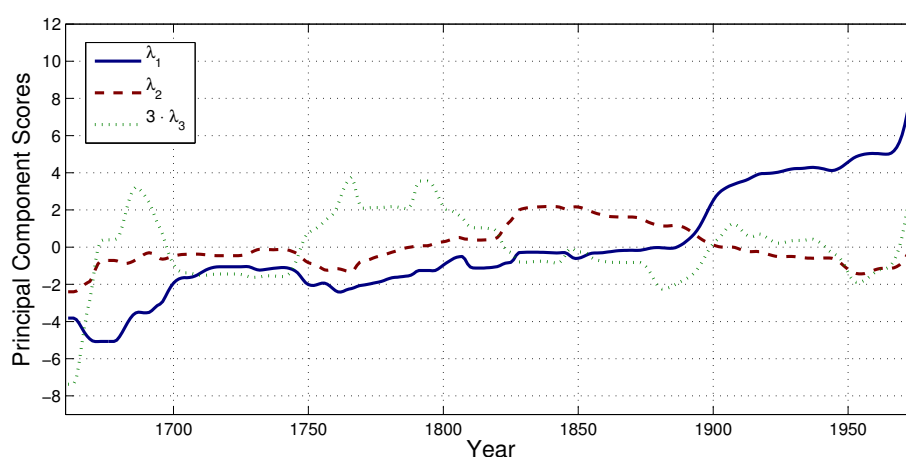


Figure 7.16. First three principal components from interval and complexity features. We show the values of the PCA scores for the individual years. For better recognition, we re-scaled the third component $\lambda_{3,i}$ with a factor of 3.

clusters. With the optimal number of six clusters, the Classical era constitutes one cluster and the Baroque time is split at about 1700. Sometimes, a cluster comprises years that are not continuously connected. As an example, we find for $K_{KM} \in [3 : 7]$ the same cluster assignment for the first years (1660–1680) and the Classical period. In this case, this may not be a meaningful observation since only few composers contribute to the first years.

7.3.1.2 Interval and Complexity Features

To compare such clustering results for other types of features, we perform the same experiments on the features used in Section 7.2.3. On the basis of NNLS chroma features, we consider the six simultaneous interval types ($\Psi^{IC1}, \dots, \Psi^{IC6}$) as well as the complexity feature Γ_{Fifth} on four different time scales (10 features, in total). Before performing PCA, we normalize the rows by subtracting their mean value. Furthermore, we have to divide the rows by their standard deviation since intervals and complexity are different types of features with individual scales. In Figure 7.16, we show the resulting PCA scores. Table 7.3 lists the entries of the associated weight vectors. The first component increases over the years and particularly marks the stylistic change at about 1900. Looking at the entries of \mathbf{w}^1 , we see

Table 7.3. Principal component weights for interval and complexity features. The interval features rely on local chroma features ($\text{NNLS}_{\text{local}}$). For the complexity, we selected the feature Γ_{Fifth} based on four different time resolutions.

Feature type	w^1	w^2	w^3
Ψ^{IC1}	.341	-.140	.081
Ψ^{IC2}	.334	-.128	-.287
Ψ^{IC3}	-.087	.881	-.363
Ψ^{IC4}	-.292	.204	.739
Ψ^{IC5}	-.310	-.265	-.424
Ψ^{IC6}	.336	.197	.149
$\Gamma_{\text{Fifth}} \text{NNLS}_{\text{global}}$.335	.174	-.047
$\Gamma_{\text{Fifth}} \text{NNLS}_{100}^{200}$.344	-.031	.009
$\Gamma_{\text{Fifth}} \text{NNLS}_5^{10}$.347	.011	.132
$\Gamma_{\text{Fifth}} \text{NNLS}_{\text{local}}$.344	.077	.110

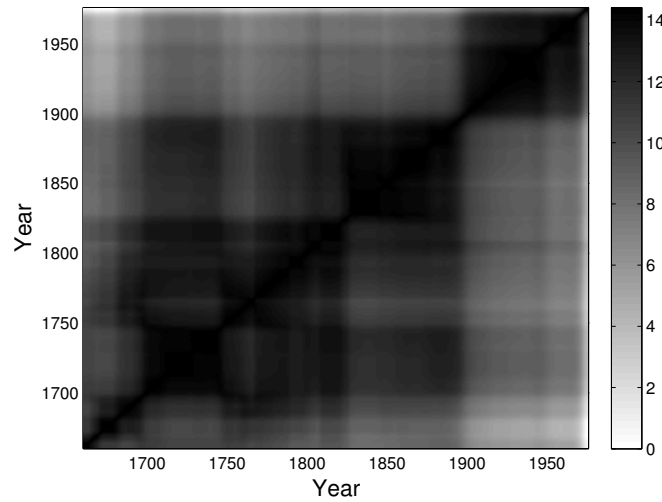


Figure 7.17. Self-similarity matrix of interval and complexity features. After normalizing the rows to mean zero and standard deviation one, we computed the self-similarity matrix based on the Euclidean distance of interval and complexity features.

that almost all dimensions have a similar weight, which may be an effect of the normalization. The entries for the complexity features all have positive sign indicating a close relationship between the first principal component and the complexity of the music, which increases over the years. The w^1 entries of the interval features support this assumption since the dissonant intervals (IC1, IC2, and IC6) have positive sign whereas the consonant intervals (IC3, IC4, and IC5) contribute with negative sign. For the second principal component, the situation is less clear. Looking at w^2 , this component seems to describe the relation between thirds—in particular, minor thirds with a weight of .881—and other intervals such as perfect fifths (IC5 with a negative sign). From Figure 7.16, we see that this component mainly discriminates the Romantic period (about 1825-1890) from the other years. This might point to the observation that chords with many third intervals—such as seventh or ninth chords—are important for Romantic styles. The positive coefficient of the tritone in w^2 indicates an important role of diminished and half-diminished triads. For the third principal component, the relation between major and minor thirds seems to be crucial since IC3 and IC4 have large values with opposite sign. The score of this component suggests a relation to the Classical period.

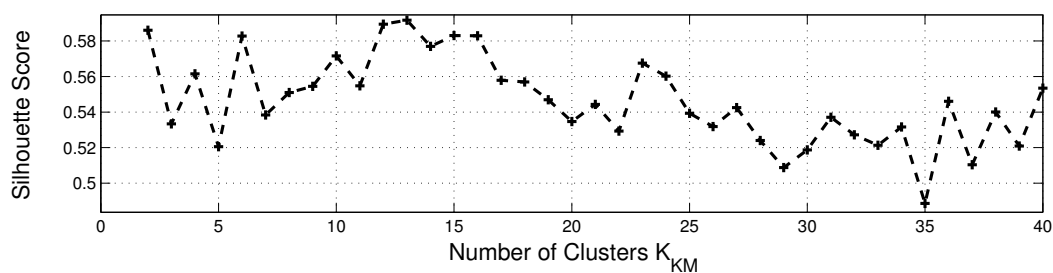


Figure 7.18. K-means clustering based on interval and complexity features. On the first three principal components, we repeat the clustering procedure 200 times and calculate the silhouette scores. We display the mean scores over the number of clusters K_{KM} .

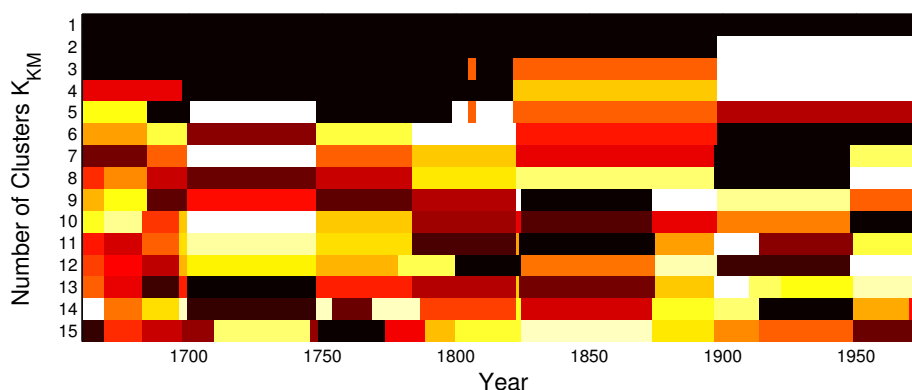


Figure 7.19. Clustering of years for local chroma-based features. This plot illustrates the clusters of years for a different number of clusters K_{KM} . Each cluster obtains a different color.

Figure 7.17 shows the self-similarity matrix for these features. Since interval and complexity features have different scales, we calculated this matrix on the basis of three principal components (after normalization). The two main homogeneous blocks separate around 1900. This may indicate that complexity and interval features are useful to distinguish tonal and atonal music. We observe further structures before the year 1900, which are less obvious.

To obtain a meaningful number of clusters, we calculate the silhouette score for the K -means algorithm on the basis of the interval and complexity features. We only use the first three principal components as input. In this scenario, we find an optimal number of clusters of 12 or 13. The optimal value obtained for the root progressions—six clusters—also has a high silhouette score here. Looking at the clusters (Figure 7.19), we find stable cluster boundaries at about 1900 and 1700, similar to Figure 7.15. The first border arises at 1900, which mainly seems to discriminate tonal from atonal pieces. The boundary at 1750 arises for $K_{KM} \geq 5$ clusters and, thus, seems to be less obvious than for chord progressions (Figure 7.12). In contrast, the 1820 boundary seems to be more important when using intervals and complexity. Furthermore, there is a boundary at 1780. The boundary at 1800—observed for the chord progressions—does not play a major role here. Principally, the clustering result is different with other types of features. Nevertheless, some change points in music history (for example, at 1750 or 1900) seem to establish with both feature types independently from each other.

7.3.1.3 Feature Combination

In the previous sections, we saw that chord progression statistics and local chroma-based features may complementarily capture different aspects of stylistic similarity. For this reason, we

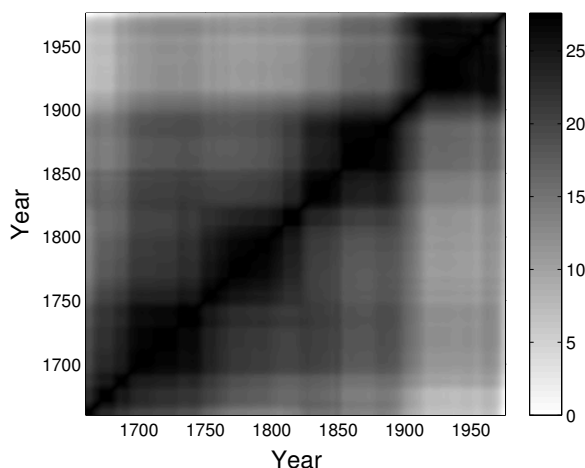


Figure 7.20. Self-similarity matrix based on the feature combination. Using the first three principal components from 55 root note progressions, six interval and four complexity features, we visualize the distances between years.

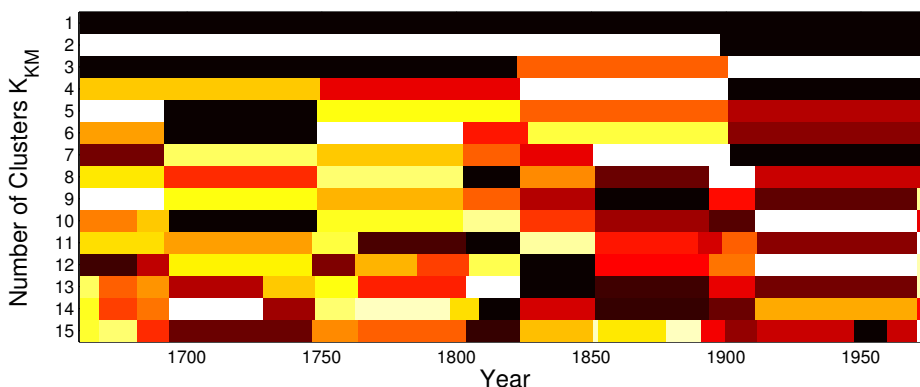


Figure 7.21. Clustering result for a combination of features. Based on the first three principal components from all features, we plot the cluster assignment of the years for different numbers of clusters.

combine both feature types in the following. To add more detailed information about chord progressions, we now consider the specific root note progressions with respect to the chord types (major / minor) as presented in Section 7.2.2.¹⁵ Leaving out the self-transitions, we end up with $11 \times 5 = 55$ dimensions of root note progressions (compare Figure 7.7). Together with the ten interval and complexity measures from Section 7.3.1.2, we have 65 feature dimensions in total. On this data, we perform PCA with a prior normalization of the rows to a mean of zero and a standard deviation of one. For the first three principal components, we compute a self-similarity matrix using the Euclidean distance (Figure 7.20). Comparing this plot to Figures 7.12 and 7.17, we find influences from both features. The clear separation at roughly the year 1900 probably stems from the interval and complexity features. Furthermore, these features seem to contribute to some homogeneity of the Romantic era (about 1820–1900). In contrast, the splitting into two sub-blocks at 1850 may result from the chord progressions since this is a major boundary in Figure 7.12. When ignoring the years before 1700 (few composers contributing), we find a division into four main eras with several sub-structures.

¹⁵In Section 7.3.1.1, we only used root note progressions independently from the chord types in order to enable an easier interpretation of the results.

In Figure 7.21, we show the result of the K -means clustering algorithm on the basis of the first three principal components. As we expect from the structure of the self-similarity matrix, the years 1750 and 1900 play a major part for separating clusters. Similar to the interval and complexity features (Figure 7.19), the boundary at 1820 seems to be important whereas the 1850 boundary—indicated by the chord progressions in Figure 7.15—only appears for seven or more clusters. The Baroque period separates into two clusters at 1700 for $K_{KM} \geq 5$. Clustering into six or more clusters, we find at least one “intermediate period” between the Classical and Romantic eras. In summary, the clustering results based on the feature combination seem to be a bit smoother than for the individual feature types. Most of the boundaries between clusters coincide with breaking points proposed by music historians. Nevertheless, a clustering of years with several contributing composers cannot resolve details of stylistic evolution, which often exhibits parallel and contrasting trends. As we mentioned in Section 4.6.2, Rodriguez Zivic *et al.* [202] performed a similar clustering of years based on melodic intervals from symbolic data (the “Peachnote” corpus [247]). Though they have the exact composition dates in their dataset—in contrast to our scenario—the results may be comparable to some degree since they use a smoothing window of ten years for the clustering results. As a result, they obtained roughly similar break points between their four clusters—at the years 1760, 1825, and 1895. This is a very interesting agreement since they derived their features from score data using Optical Music Recognition—a completely different type of data. For this reason, we might be willing to assume that our clustering methods uncover some historical evolutions of style even though the features themselves and the clustering procedure may be error-prone and inaccurate on the fine level.

7.3.2 Clustering Individual Pieces

To better account for the stylistic inhomogeneity of the years, we perform our clustering experiment with an inverted order. We consider the combined features as used in Section 7.3.1.3 (55 chord progression, six interval, and four complexity features) for the *individual pieces* without prior mapping to years. On the resulting feature matrix $\mathcal{F}_{\text{Pieces}} \in \mathbb{R}^{65 \times 2000}$, we perform PCA after normalizing the rows to a mean of zero and a standard deviation of one. To the reduced matrix (three principal components), we apply the K -means algorithm with a number of $K_{KM} = 5$ clusters.

With this procedure, we assign every piece in the dataset to one of the five clusters. As the next step, we map the *cluster assignments* of the individual pieces onto the time axis with the procedure shown in Section 7.2.1. The resulting distribution describes the fraction of pieces belonging to each cluster over the years. In Figure 7.22, we individually show this fraction for the five clusters. Figure 7.23 jointly visualizes all cluster assignments as stacked bars.

We now want to discuss the possible meanings of the different clusters. Compared to the previous sections (clustering of years), the results are much less clear. Cluster 1 exhibits the most descriptive distribution. This cluster enters gradually during the 19th century and seems to play an important role in the 20th century. We assume that this is the “Avant-garde” cluster, which is mostly characterized by complex and atonal pieces. Nevertheless, this is not the only cluster present in the 20th century. Cluster 5 also contributes here, which is the most prominent cluster throughout the 19th century (“Romantic” cluster) but also shows influence in previous years. The presence of cluster 1 (“Avant-garde”) and cluster 5

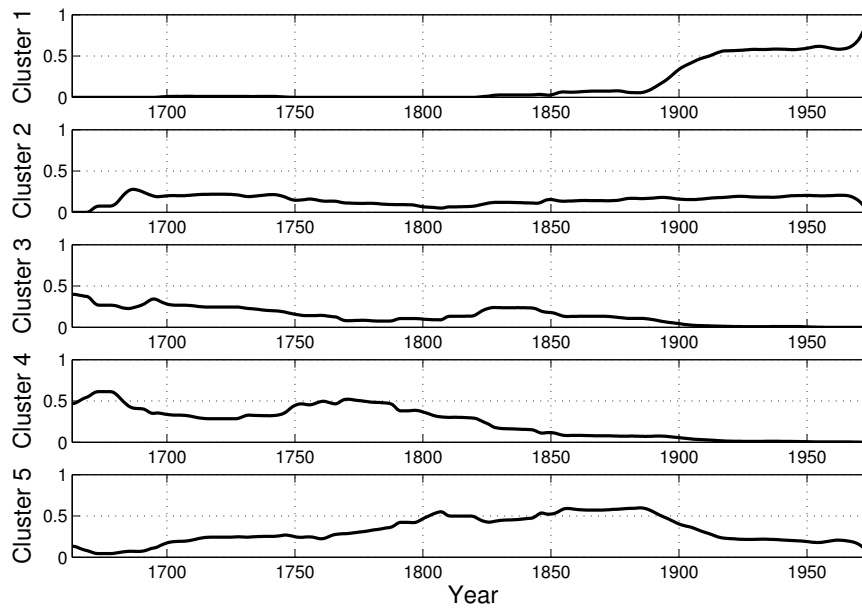


Figure 7.22. K-means clustering of individual pieces distributed over the years. For a fixed number of $K_{KM} = 5$ clusters, we assign every piece to a cluster. Mapping the assignments over the years, we obtain the fractions of pieces per year that belong to each of the individual clusters. Figure 7.23 shows a color plot of the same values.

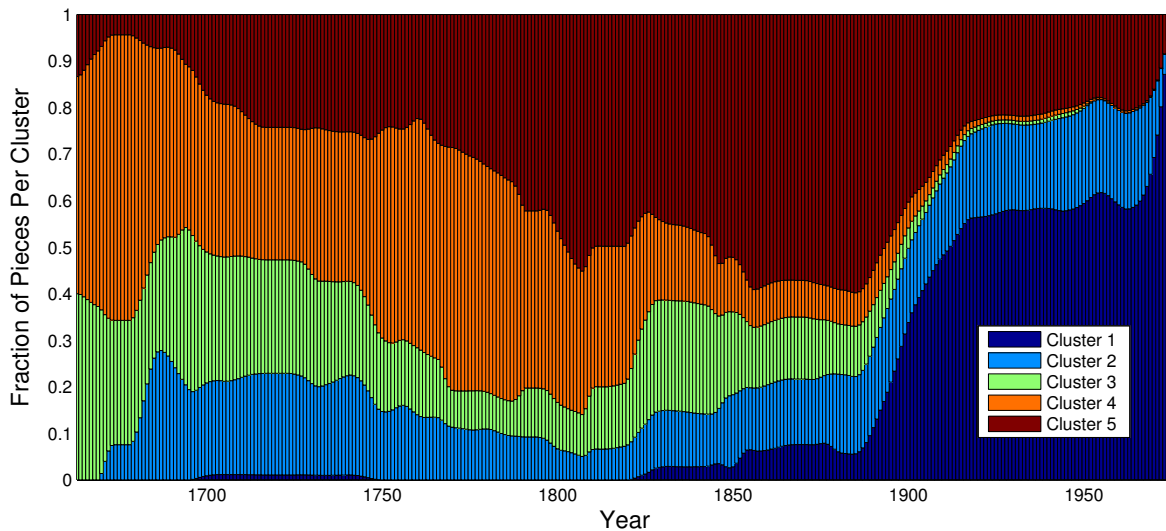


Figure 7.23. K-means clustering of individual pieces as bar histogram. The fraction of pieces belonging to each cluster is indicated by bars in different colors.

(“Romantic”) for the years 1910–1960 may reflect the parallelism of styles during this time.¹⁶ Cluster 2 is also present during the 20th century. This cluster obtains a very flat distribution over the years so that we can hardly interpret its meaning (“noise cluster”). The meaning of the clusters 3 and 4 is not very clear. They seem to mostly describe the Baroque and Classical periods and decrease to small values after 1850. Hereby, cluster 3 is slightly more prominent for the Baroque time and has less contributions to the years 1750–1820 (Classical

¹⁶For example, romantic pieces by R. Strauss and dodecahonic music by A. Schönberg simultaneously contribute here.

period). After this, we see a small “revival” of this cluster for 1820–1850. As opposed to this, cluster 4 is more important for the early classical time (1750–1800). Nevertheless, this cluster also contributes to the Baroque period and, in particular, to the years before 1700.

We see that the situation is much less distinct when clustering pieces before mapping to years. None of the cluster covers more than 60 % of the pieces for a considerable span of time. The individuality of pieces and composers seems to be stronger than the stylistic homogeneity of a period. This indicates that the procedure of the previous sections (clustering years after averaging) has some limitations. Supposedly, first averaging over all piece-wise features of a year—followed by clustering the years—is too superficial and obscures the heterogeneity of the pieces contributing to a year. Though being ambiguous to some degree, the clustering of pieces before mapping to years provides some insights into historical trends.

7.3.3 Clustering Composers

Finally, we want to use our methods to analyze the stylistic relation between different composers. For each of the 70 composers, we average the chord progression, interval, and complexity features over all pieces by the respective composer. On the resulting feature matrix $\mathcal{F}_{\text{Composers}} \in \mathbb{R}^{65 \times 70}$, we perform PCA followed by K -means clustering on the first three principal components. We choose a number of $K_{\text{KM}} = 5$ clusters.

In Figure 7.24, we display the resulting cluster assignments for the composers as colored bars of their lifetime. This plot relates to the overview plot in Figure 7.1—but here, the colors indicate the automatic cluster assignments instead of the annotated classes. The results seem to be very interesting. Mostly, composers with a similar lifetime belong to the same cluster. This indicates some fundamental relationship between historical context and stylistic similarity. For example, Cluster 1 (green) comprises most of the Baroque composers. However, single composers escape such a simple partitioning. For example, A. Vivaldi and D. Scarlatti obtain the cluster label of the Classical time. If we try to find musical reasons for this attribution, we might argue that the harmonic properties of A. Vivaldi’s music show some similarities with music from the Classical period. As another interesting observation, C. P. E. Bach belongs to the Romantic cluster. Often, musicologists label his music as the “sensitive style” (“Empfindsamer Stil”)—one of the pre-classical trends, which was indeed motivated by some ideas that relate to Romantic paradigms. For the other two composers assigned to the Romantic cluster in this time (L. Giustini and G. B. Platti), we are not aware of such relations. As for this example, such kind of rather surprising observations could be a starting point for musicological research in the future. Other pre-classical composers such as J. Stamitz, L. Mozart, or J. C. Bach belong to the Classical cluster. For the stylistic change between the Classical and Romantic periods, we find a rather clear separation. Here, L. van Beethoven, C. M. von Weber, and G. Rossini constitute the latest Classical representatives whereas F. Schubert and F. Mendelssohn belong to the Romantic cluster. For the 20th century, we find two parallel clusters. The yellow cluster (Cluster 5) comprises the avant-garde of that time with mostly rigorous atonal composers such as A. Schönberg, A. Berg, A. Webern, I. Stravinsky, E. Varèse, or B. Bartók. Furthermore, the younger 20th century composers B. Britten, O. Messiaen, and P. Boulez belong to this cluster. The other modern cluster (Cluster 4, red) contains composers with a more moderate harmonic style such as S. Prokofiev and D. Shostakovich. The assignment of M. Mussorgsky and G. Faure to this cluster is rather surprising since most of the late romantic composers (G. Mahler, R. Strauss) as well as the impressionists (C. Debussy, M. Ravel) belong to the Romantic cluster.

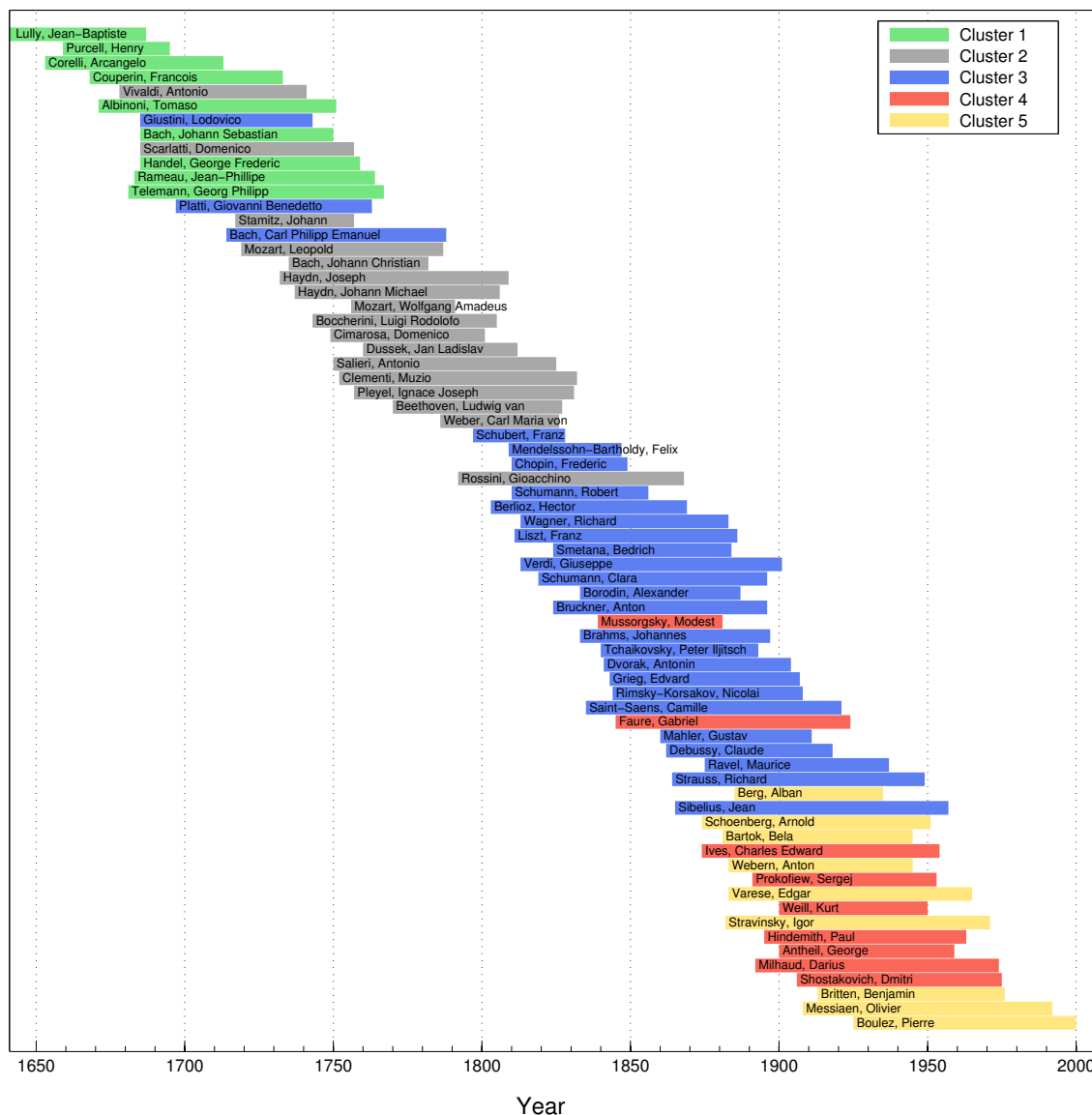


Figure 7.24. K-means clustering of composers. With a fixed number of $K_{KM} = 5$ clusters, we assigned the pieces of each composer to one of the clusters. Here, we show the lifetimes of the composers with the color indicating the cluster assignments.

These outliers point to the difficulties of clustering composers to a fixed number of top-level clusters. As an outlook, we therefore present two studies of applying methods for *hierarchical clustering* to such type of features. In bioinformatics, these phylogenetic trees are popular tools for clustering DNA sequences in order to highlight evolutionary developments and trends. The trees rely on the Euclidean distance between feature vectors and hierarchically arrange composers into similarity groups of variable size. Figures 7.25 and 7.26 show two of these phylogenetic trees computed with different configurations. In Figure 7.25, the two main groups—divided at the first node—roughly relate to tonal and atonal composers. Most of the composer pairings seem to be stylistically meaningful such as W. A. Mozart – J. Haydn, C. Debussy – M. Ravel, or R. Schumann – F. Mendelssohn. But there are also limitations.

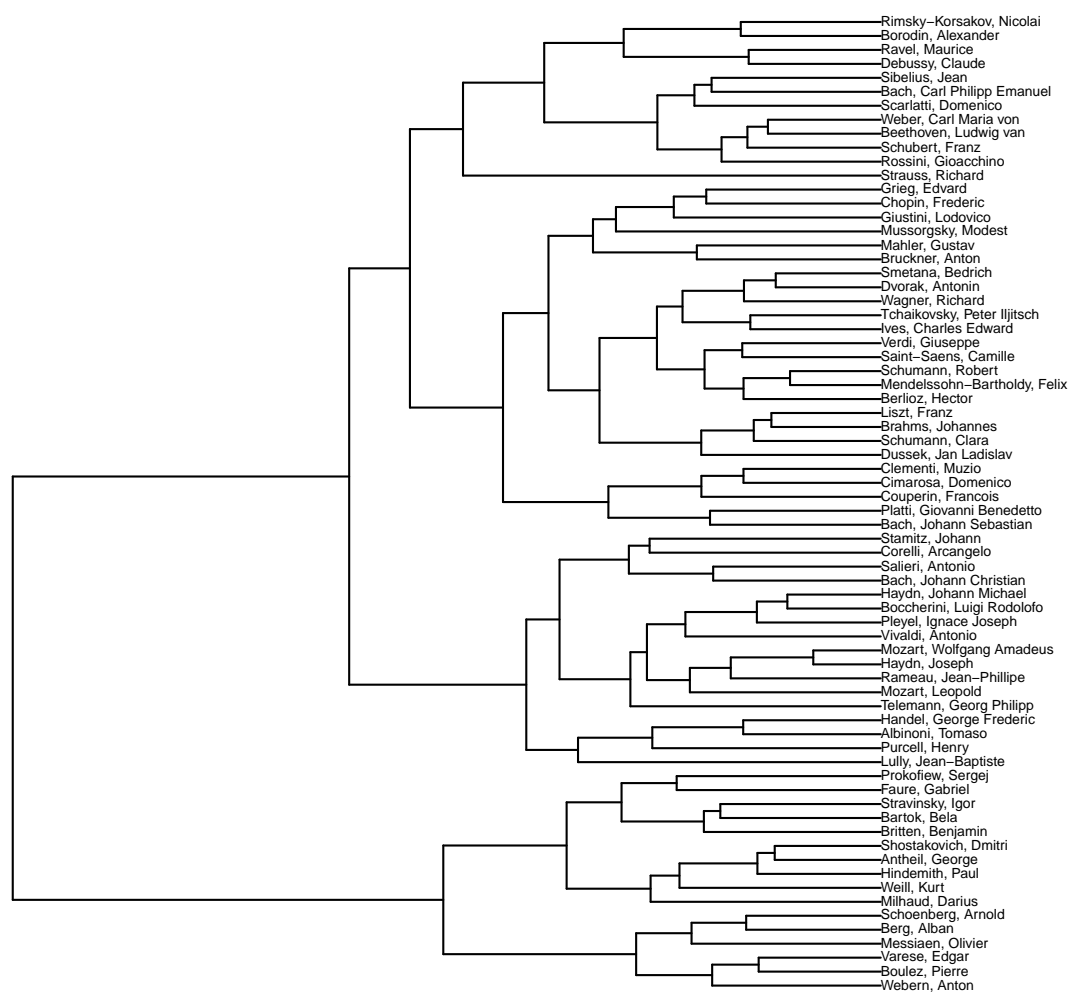


Figure 7.25. Hierarchical clustering of composers. This phylogenetic tree method relies on the maximal distance of individual elements in the two initial clusters (complete-linkage clustering).

For example, the pairing of J. Sibelius with C. P. E. Bach does probably not reflect meaningful stylistic similarity.

In Figure 7.26, we show a tree computed with a different method. Here, the branch length provides further information about distances between items. The total branch length from one composer to another corresponds to their “stylistic distance.” Interestingly, the horizontal position of the composers seems to roughly correlate to some kind of “tonal evolution” of their music. Most Baroque composers stand at the very left side whereas the group to the very right comprises the atonal composers. Though they are far from being a “final statement” about any stylistic relationship between composer, such feature-based methods seem to provide meaningful insights about the interrelation of composer styles—even beyond the well-known connections.

7.4 Conclusion

In this chapter, we applied visualization and clustering methods for exploring stylistic and historic relations within Western classical music. We presented our dataset, which comprises



Figure 7.26. Hierarchical clustering using the minimum evolution criterion. With this method (ordinary least squares with unweighted subtrees), the branch length corresponds to the distance of nodes or items to their ancestor. The minimum evolution criterion enforces the total length of branches to be minimal.

2000 audio recordings of piano and orchestra pieces by 70 composers and from almost 400 years of music history. From these recordings, we automatically extracted audio features for describing tonal structures. The first type of features serves to quantify chord progressions. From the estimated chord sequences, we derived statistics of progressions with respect to the chords' root notes. The second class of features aims at quantifying the presence of interval types and the degree of tonal complexity—as introduced in Chapter 6.

In the first step, we mapped these features onto a historical time axis regarding the lifetime of the composers. These visualizations of the features showed interesting trends, which, to a certain extent, seem to describe stylistic evolutions in the dataset. For example, we observed an increasing use of major and minor thirds during the 19th century—both as distances of chord progressions and as simultaneous intervals. Furthermore, a higher frequency of authentic chord progression compared to plagal progressions occurred as a decisive feature to discriminate late baroque music from classical style pieces. The tonal complexity features showed a minimum in the classical period and increased during the 19th century and, in particular, towards the 20th century. Interestingly, this increase was stronger regarding global complexity (referring to the arrangement of keys on a large scale) than for local

complexity values (referring to the structure of chords). For most of the features, the change at about 1900 was the most remarkable one—primarily caused by the atonal music arising at that time. Fucks and Lauter [66] reported similar findings in their statistical analyses of instrumental parts from score data.

Next, we performed several clustering experiments with respect to years, individual pieces, and composers. Though not all relations highlighted by the clustering are musically meaningful, we made several observations that confirm common assumptions of stylistic trends in music history. Furthermore, some of our findings point to rather unknown stylistic relationships between pieces or composers. For clustering *years*, we found groups that primarily correlate to the historical periods as commonly outlined by musicologists. It is an encouraging result to obtain this typical partitioning of periods from automatic data analysis without any prior assumptions. Interestingly, Rodriguez Zivic *et al.* [202] obtained a quite similar result using a completely different strategy (analysis of *melodic* intervals based on *graphical scores* using OMR).

Applying a different strategy—first clustering *pieces* and then mapping the clustering results to years—revealed that pieces within a period may fundamentally differ from each other. This may let us conclude that the individuality of a single piece is of greater importance than the stylistic homogeneity within a period. A possible explanation could be the hypothesis that composers usually aim at writing “novel” pieces whereas style relates to secondary characteristics that rather unconsciously “happen” in the composition process (compare the discussion in Section 2.10).

Surprisingly, we found a different picture when averaging over all works of a composer and then clustering the *composers*. In this scenario, composers living at the same time predominantly appeared in the same cluster—with two parallel “Modern” clusters in the 20th century. Based on this observation, we suppose that averaging over many works by a composer balances out the individual pieces’ characteristics (which may strongly differ between the pieces) and, thus, helps to uncover the composer’s style to a certain extent. Altogether, we assume that such kind of methods may provide useful tools for analyzing and highlighting stylistic relationships between musical works. In future studies, these analyses could support hypotheses about style evolution or point to interesting coherences that are yet to discover.

8 Subgenre Classification for Western Classical Music

In the previous chapter, we presented a couple of methods for analyzing corpora of classical music on the basis of tonal audio features. We showed that such techniques may be useful to look into subtle differences and evolutions between pieces, composers, and years. Furthermore, we presented clustering techniques to obtain an automatic grouping into different categories without prior assumptions about musical similarity. In contrast to that, we now want to approach the task of automatic classification. Such methods are called “supervised” since we train some kind of classifier on given *training data* with corresponding *class labels* (see Section 3.6). This section is mainly based on previous publications [256,258] but further provides additional experiments and more profound discussions.

For classification tasks, the structure of the dataset and the class assignments are of major importance. The data should contain a sufficiently high number of items which are representative for each class in order to enable a successful training procedure. In Section 8.1, we introduce the datasets used for our classification experiments. In this thesis, we deal with two scenarios. First, we are interested in the assignment of pieces to stylistic periods or eras. Second, we perform experiments to identify the composer of a piece. In Section 8.2, we discuss the importance of dimensionality reduction as a preprocessing step for classification. We show that dimensionality reduction may also provide interesting visualizations of the data based on the structure of the feature space. Next (Section 8.3), we present the classification results. Section 8.3.1 outlines the main experimental procedure. In Section 8.3.2, we test different classifiers and configurations on the two datasets. For experiments using cross validation, we need to ensure that no correlations exist in the data between the semantic (“musical”) properties which we want to classify and other characteristics—such as irrelevant timbral properties or artifacts from recording conditions or audio downmixing. To this account, we apply different filtering strategies for the cross validation procedure (Section 8.3.3).

As a central motivation for performing classification experiments, we want to investigate the efficiency of different feature types for recognizing style. From the visualizations presented in Chapters 6 and 7, we obtained a rough impression of these features’ “musical meaning.” We suppose that the performance of different features in classification experiments may provide some insights how important the related *musical* phenomena are for discriminating styles. Let us consider an example. If the accuracy for classifying pieces regarding the classes “Classical” and “Romantic” benefits from the use of tonal complexity features on a global scale, this might point to a high importance of modulations and global tonality for discriminating these styles. In Section 8.3.4, we draw such comparisons between different feature types. Finally (Section 8.3.5), we exemplarily look at some individual pieces in order to get a better understanding of the classification mechanisms in our systems. Section 8.4 concludes this chapter with a discussion of the benefits and problems with such classification experiments.

8.1 Datasets

As we mentioned in Chapter 1, automatic classification of music recordings into *genre* categories constitutes a main research task in the field of Music Information Retrieval. Typical classification scenarios deal with several top-level genres such as Rock, Pop, or Jazz (see Section 4.6). In this thesis, we are interested in classifying *subgenres* of classical music and in understanding the musical meaning of such categories—as discussed in Section 2.10. To this end, we compiled two datasets, each for a specific task.

The first scenario deals with the classification into historical or, more precisely, stylistic periods (“eras”). We consider the four periods Baroque, Classical, Romantic, and Modern. The Modern category contains music from the early 20th century that clearly applies advanced concepts of tonality. Typical examples for this type of music are the dodecaphonic pieces by Schönberg and his followers. As we outlined in Section 2.10, such a categorization is quite superficial. Musicologists often prefer a more detailed view considering individual composers or even single works in order to observe subtle stylistic differences. Beyond these details, one may detect more general development lines in music history as well as the breaking of such lines. This is why a classification into eras can be helpful as a first analysis step, which may precede a closer look at individual tendencies of style [65, 74, 250].

To study this scenario, we compiled a dataset with a balanced number of 400 pieces for each of the four periods. We already presented and discussed this *Cross-Era* dataset in Section 7.1. The compilation comprises works by various composers from different countries in each class. To investigate dependencies on timbral characteristics, we only included orchestra recordings on the one hand and piano recordings—played on a modern grand piano—on the other hand (no harpsichord for the Baroque class). We did not include any works featuring singing voices or the organ. Each of the four classes contains 200 orchestra and 200 piano recordings. This enables us to create the balanced subsets *Cross-Era-Piano* and *Cross-Era-Orchestra*, which might be useful to investigate timbre-invariance of the classification algorithm. Table 8.1 gives an overview of the different datasets. We avoided to include transitional composers who cannot be assigned clearly to one of the periods (such as, for example, L. van Beethoven or F. Schubert, who could be considered both as late classical or early romantic composers). To preserve the variety of movement types with respect to properties such as rhythm and mood (major/minor keys, slow/fast tempo, duple/triple meter), we included all movements or parts for most of the work cycles. For further details of the *Cross-Era* set, we refer to Table 7.1 and Figure 7.1.

In the *Cross-Era* set, we summarized several composers into one stylistic class, respectively. To go beyond this simplified scenario, we also approach the problem of *composer identification*. Moreover, this task allows for a better comparison to state-of-the-art algorithms since the composer identification problem was approached more often.¹ For these reasons, we compiled another dataset comprising 100 pieces by each of the eleven composers J. S. Bach, L. van Beethoven, J. Brahms, A. Dvořak, G. F. Handel, J. Haydn, F. Mendelssohn Bartholdy, W. A. Mozart, J.-P. Rameau, F. Schubert, and D. Shostakovich. Here, we included a large variety of instrumentations including—among others—orchestral works, piano pieces, and solo concertos as well as compositions for choir, organ, and harpsichord. The pieces stem from commercial recordings on 94 different albums and are played by 68 different interpreters. Table 8.2 provides more detailed information about the dataset.

¹For example, the annual evaluation contest MIREX for MIR algorithms includes a composer identification task with eleven composers (<http://www.music-ir.org/mirex/>).

Table 8.1. Classification datasets and their properties. From the two main datasets *Cross-Era* and *Cross-Composer*, we compiled different subsets.

<i>Dataset</i>	<i>Classes</i>	<i>No. classes</i>	<i>Items per class</i>	<i>Total items</i>
<i>Cross-Era-Full</i>	Baroque; Classical; Romantic; Modern	4	400	1600
<i>Cross-Era-Piano</i>	Baroque; Classical; Romantic; Modern	4	200	800
<i>Cross-Era-Orchestra</i>	Baroque; Classical; Romantic; Modern	4	200	800
<i>Cross-Comp-11</i>	Bach, J. S.; Beethoven, L. van; Brahms, J.; Dvořak, A.; Handel, G. F.; Haydn, J.; Mendelssohn Bartholdy, F.; Mozart, W. A.; Rameau, J.-P.; Schubert, F.; Shostakovich, D.	11	100	1100
<i>Cross-Comp-5</i>	Bach, J. S.; Beethoven, L. van; Brahms, J.; Haydn, J.; Shostakovich, D.	5	100	500

Table 8.2. Cross-Composer dataset. The percentage numbers indicate the fraction of works featuring the instruments. Here, we only mention the more frequent orchestrations.

<i>Instruments</i>	<i>Fraction of Pieces</i>
Orchestra	38.7 %
Piano	38.6 %
Ensemble	19.5 %
Choir	6.6 %
Organ	6.3 %

To enable a comparison with the MIREX results, we chose the same number of 11 composers. Due to our data resources, we did not use exactly the same composers but replaced F. Chopin and A. Vivaldi with J. P. Rameau and D. Shostakovich. In contrast to the MIREX data, which contains audio excerpts of 30 seconds length, we use the full-length tracks for our classification experiments.² We made features and annotations for this dataset publicly available on a website.³

Since an eleven-class problem is quite a challenging task for any classification algorithm, we further make use of a subset. To this end, we selected five of the composers that are stylistically more distinct from each other than is the case for the full dataset (see lower part of Table 8.1). In the following, we refer to the full dataset as *Cross-Comp-11* and to the reduced one as *Cross-Comp-5*.

²In our opinion, it is musically more meaningful to use full-length recordings (movements). For example, we may perceive an excerpt from the development phase in a Mozart symphony movement as stylistically different from an excerpt from the exposition of the same movement. Furthermore, global characteristics such as repetitions, modulations, and formal aspects may constitute relevant stylistic cues (compare Section 2.10 and [129]), which we lose when using only 30-second clips.

³<http://www.audiolabs-erlangen.de/resources/MIR/cross-comp>

8.2 Dimensionality Reduction

As we discussed in Section 3.6, the feature dimensionality $D \in \mathbb{N}$ can be quite large. In our case, we combine different types of chroma-based features each with different configurations leading to $D > 100$ features for many scenarios. Since the size of our datasets is limited, we have to take care of the “curse of dimensionality” (compare Section 3.6.1 and [246]). To prevent overfitting due to this effect, we apply a method known as Fisher transformation or Linear Discriminant Analysis (LDA) for reducing the feature dimensionality to a smaller number $L < D$ (Section 3.6.4.2). This supervised decomposition reduces the feature dimensionality in such a way that the class separation is optimal [248]. For a scenario with Z classes, we use the maximum number of

$$L := Z - 1 \tag{8.1}$$

linearly independent dimensions [5]. Since our datasets contain at least $I = 100$ instances per class, we fulfill the common rule of thumb $I \geq 10 \cdot L$ [107, 198].

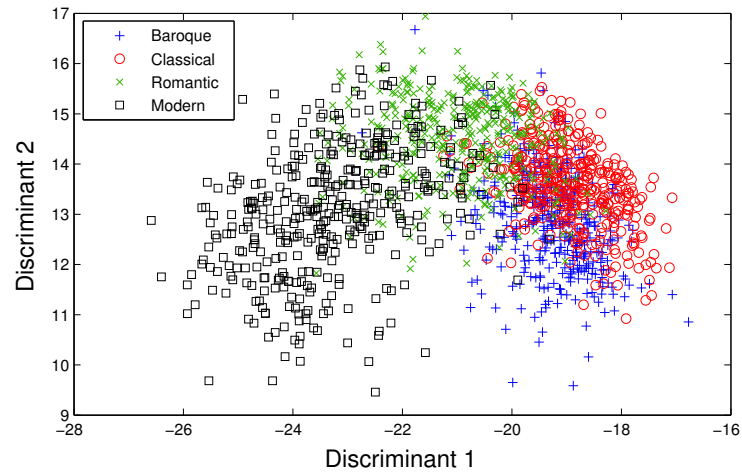
Beyond this purpose, we can also use LDA for visualization purposes. To this end, we only use $L = 2$ output dimensions and visualize the instances in a two-dimensional plot. Since LDA aims for a maximal separation of the classes, features with high discriminative power should lead to a clear visual separation. Figure 8.1 shows such plots for the *Cross-Era-Full* dataset on the basis of different types of features. For all feature configurations, the spatial arrangement of the classes is in accordance with their historical ordering (Baroque–Classical–Romantic–Modern). To a great extent, overlapping regions only occur between neighboring periods such as Classical–Romantic.

For the first plot, we used template-based features for six interval and four triad types as presented in Section 6.1.3. We derived these features from NNLS chromagrams⁴ in four different temporal resolutions ($\text{NNLS}_{\text{local}}$, NNLS_{100}^{200} , NNLS_5^{10} , and $\text{NNLS}_{\text{global}}$). From the same chroma features, we computed seven types of tonal complexity features as outlined in Section 6.2. From all these local features, we calculated the mean and standard deviation per piece ending up with $D = 2 \cdot 4 \cdot (6 + 4 + 7) = 136$ feature dimensions. Looking at Figure 8.1 a, we see that not all of the periods are separable with the chroma-based features. In particular, the separation of the Baroque and Classical classes seems to be hard. If a considerable difference between Baroque and Classical harmony exists, our features seem not to capture these characteristics sufficiently. In contrast, the discrimination of Modern against the other styles is rather clear. This indicates that interval- and complexity features can discriminate between tonal (low complexity) and atonal music (high complexity). The desired separation of the Romantic style and the Classical style may be the result of a slightly higher tonal complexity of Romantic music compared to Classical music.

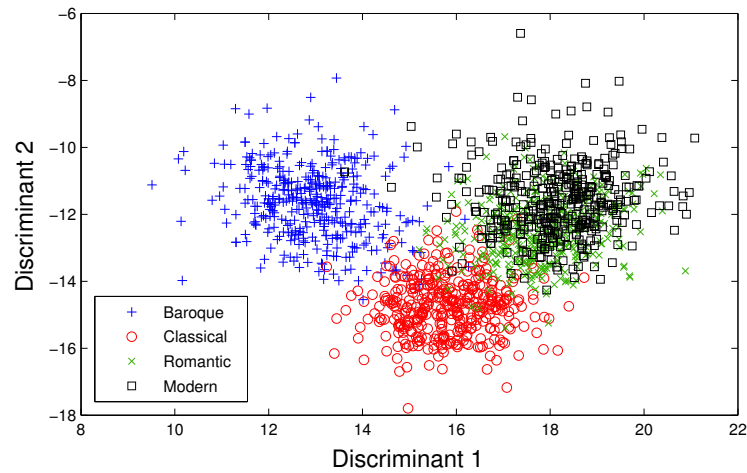
To compare our results with common methods, we also test standard audio features for calculating LDA visualizations, which we mostly calculate for several frequency bands each (see Section 3.4 for more details). We consider Mel Frequency Cepstral Coefficients (16 dimensions), Octave Spectral Contrast (14), Zero Crossing Rate (1) and Audio Spectral Envelope (16), Spectral Flatness Measure (16), Spectral Crest Factor (16), and Spectral Centroid (16). Furthermore, we use the two loudness features Θ_{LogLoud} (12) and Θ_{NormLoud} (12). Calculating mean and standard deviation over the local values results in $D = 2 \cdot (16 + 14 + 1 + 16 + 16 + 16 + 16 + 12 + 12) = 238$ features. When performing LDA using these features, we observe a different distribution of the data (Figure 8.1 b). In particular, we

⁴For a comparison of different chroma features for classification experiments, we refer to Section 8.3.4.

a) Chroma-based features (136 \rightarrow 2 dimensions)



b) Standard features (238 \rightarrow 2 dimensions)



c) Chroma-based + Standard features combined (374 \rightarrow 2 dimensions)

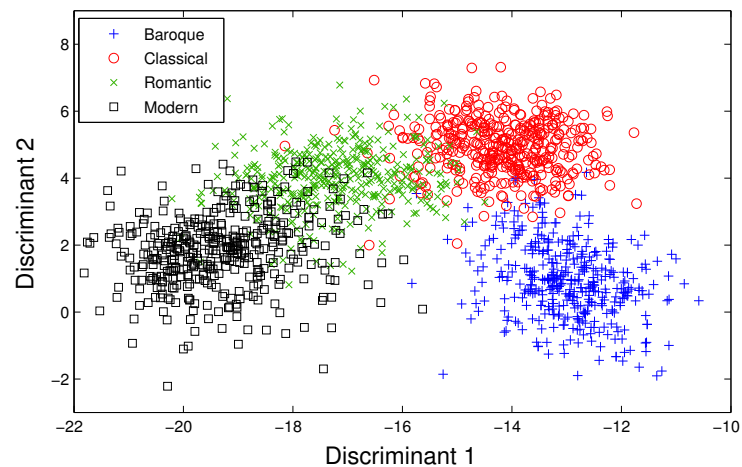


Figure 8.1. LDA visualizations of the Cross-Era-full dataset. In the upper plot (a), we performed LDA for a set of interval- and complexity features on the basis of NLS chromagrams. The middle plot (b) relies on several types of standard features. For plot (c), we combined all of these features.

obtain a good separation of Baroque and Classical pieces here. This may be the result of a considerable change between these periods regarding the instrumentation of the music. Indications for such a change may be the disappearance of the figured bass (basso continuo) in orchestral music—usually played with the involvement of a harpsichord—or a different use of octave registers due to the development of keyboard instruments. As opposed to this, we cannot really discriminate Romantic music from Classical and—even more—from Modern music with standard features. A possible reason for this may be the rather continuous evolution of instrumentation from the Classical period on. For example, the scoring of an orchestra was extended step by step from a small Classical orchestra (Haydn) to a huge Romantic orchestra (Bruckner), which most of the modern composers changed only slightly (Shostakovich). Using standard features for separating orchestra data only (Figure 8.2 d) confirms this assumption. Here, the Romantic pieces also overlap with Classical and even more with Modern pieces. For the piano case (Figure 8.2 c), Romantic and Modern pieces completely overlap. Regarding timbre, the piano almost reached its modern form and range at the beginning of the Romantic period. Therefore, the way of using the sound and range of pianos may have changed only marginally for later composers in our dataset. This might be an explanation why standard features cannot separate Romantic and Modern periods.

Due to the different behavior of chroma-based features and standard features, the separation capability may benefit from a combination of the two feature types. Figure 8.1 c confirms this assumption. Using both feature sets, we can discriminate Baroque and Classical music well thanks to the standard features. The separation between Romantic and Modern is not perfect but considerably better than for standard features alone. Discrimination of Classical and Romantic pieces also benefits from the joint usage of the features, but is still difficult. This is in accordance with musicological expectations since the stylistic change from the Classical to the Romantic period is not very distinctive.

To study the timbre-invariance of the chroma-based features, we performed LDA visualizations of the subsets *Cross-Era-Piano* and *Cross-Era-Orchestra* individually (Figure 8.2). Compared to the reduction of the full dataset using these features (Figure 8.1 a), these scenarios show slightly better separation of classes for most cases. In general, orchestral music seems to be somewhat easier to separate. Similar to the full dataset, Baroque–Classical constitutes the main problem for chroma-based features and Romantic–Modern for standard features. Combining the feature sets leads to a good separation for both *Cross-Era* subsets.

Finally, we want to apply such visualization methods to the *Cross-Composer* dataset as well. Since eleven composers are hard to display in two dimensions, we restrict ourselves to subsets with five and three composers, respectively (Figure 8.3). Let us first consider the five-composer scenario (left hand side). Here, chroma-based features do not lead to a good separation of classes in two dimensions. The pieces by Shostakovich are lying somewhat outside the region with the highest density of points. This points to a better separation of 20th century music with tonal features. Data points for the classes Beethoven and Haydn highly mix with each other and, to a smaller extent, with Bach and Brahms. With standard features, the visualization is more discriminative. Here, every composer obtains his own region. Between neighboring regions, we find some minor overlaps. With a combination, we almost achieve an identical plot as with standard features alone. Therefore, we assume that standard features are mainly responsible for the separation here. It is not clear, why the composer separation works better when including standard features. Since the classes have a rather small size and not a very broad variety, we suppose that this could be due to overfitting to the individual sound characteristic of the classes, which may be caused by a bias towards certain instrumentations or individual performers, among others.

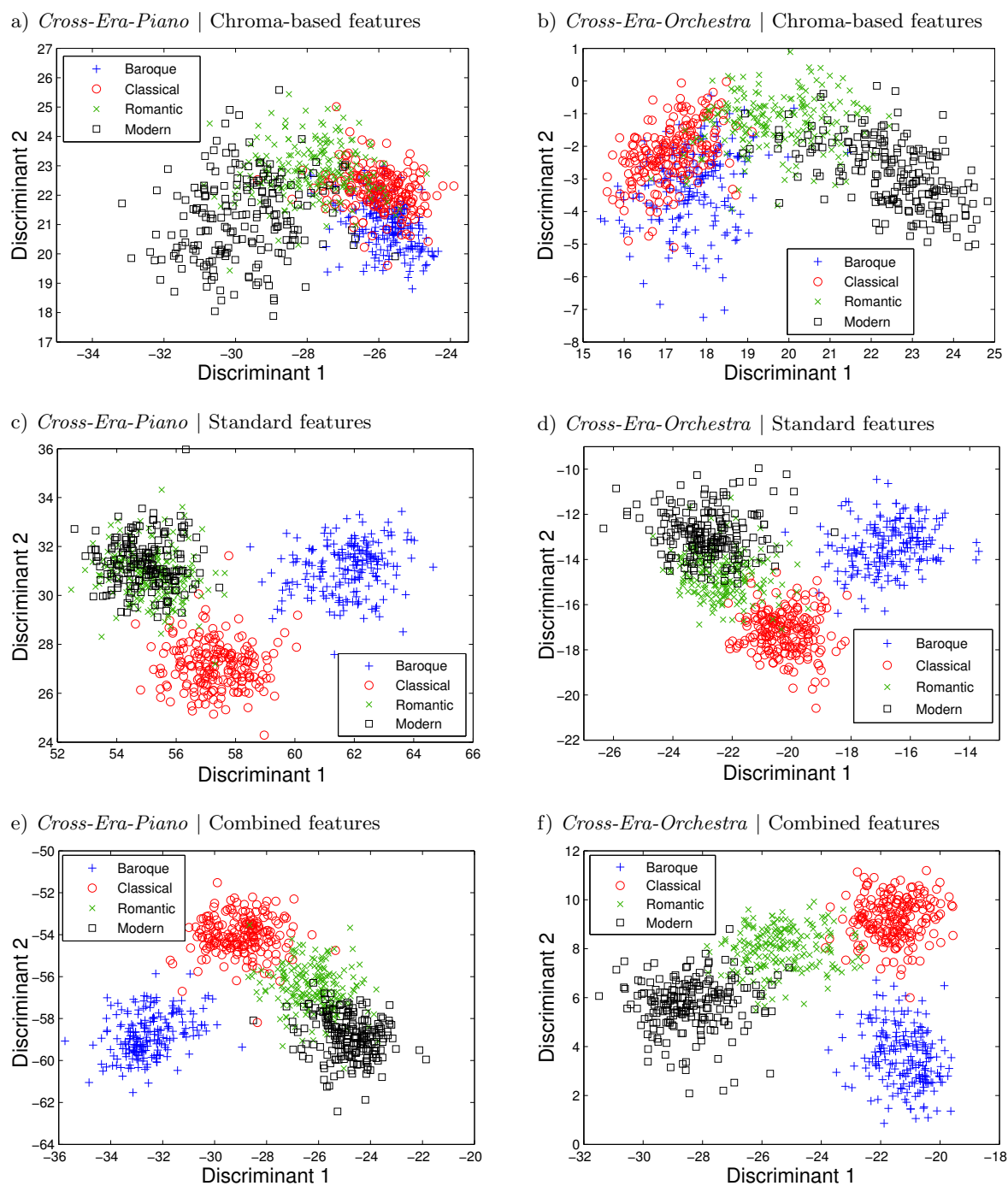


Figure 8.2. LDA visualization of the Cross-Era subsets. The left hand side shows LDA reductions of the *Cross-Era-Piano* dataset based on three different feature sets. On the right hand side, we display reductions of the *Cross-Era-Orchestra* data. The upper row (a, b) refers to chroma-based features ($D = 136$), the middle row (c, d) to standard features ($D = 238$), and the lower row (e, f) to the combination of both ($D = 374$).

For the three-composer reduction (right hand side of Figure 8.3), the plots are clearer in general. Here, we obtain a good separation with chroma-based features as well. We find slightly overlapping regions and some outliers. With standard features, the discrimination

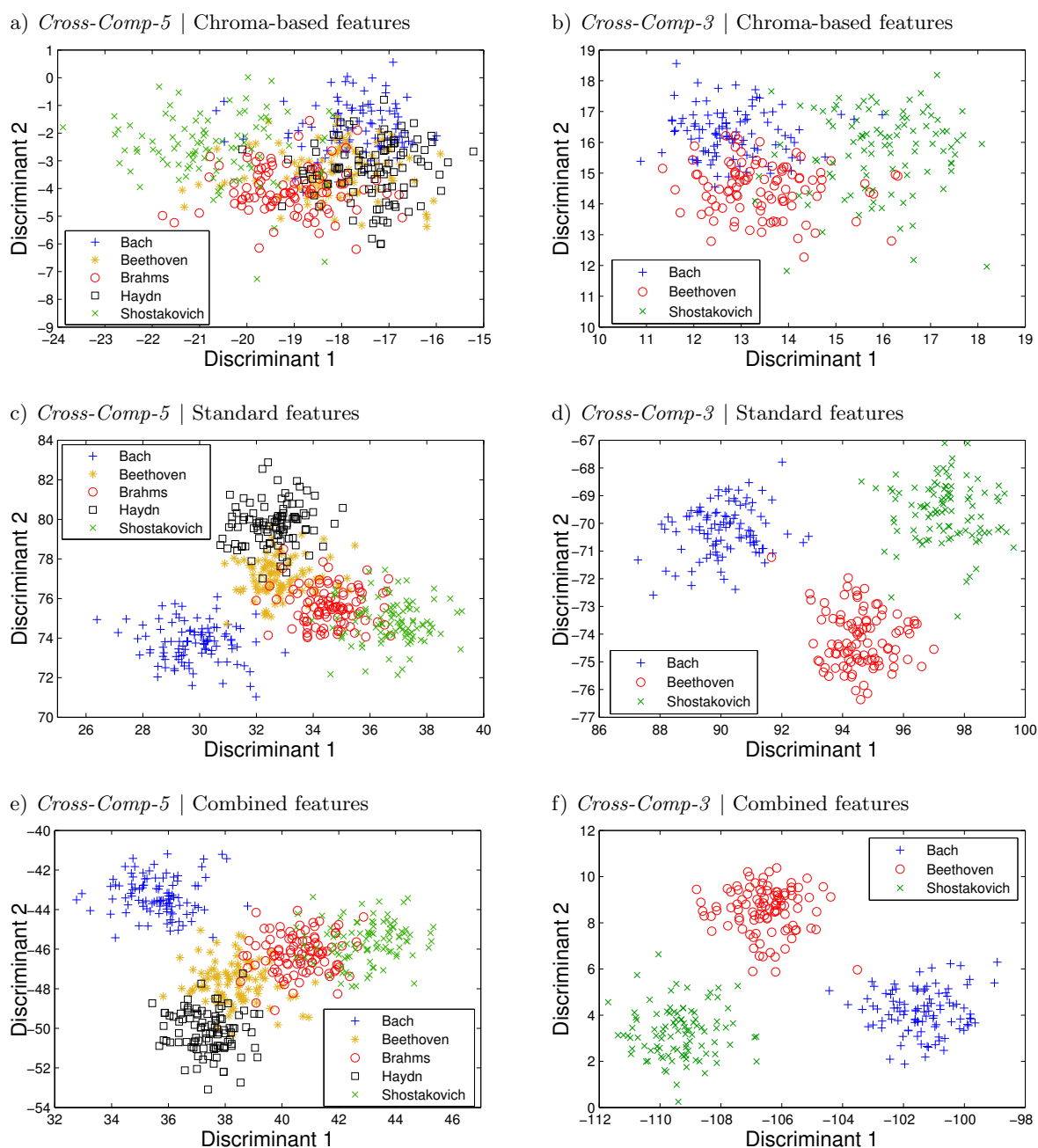


Figure 8.3. LDA visualization of two Cross-Composer subsets. On the left hand side (a, c, e), we visualize a five-composer subset whereas the right hand side (b, d, f) deals with three composers. The visualizations in the upper row (a, b) rely on chroma-based interval and complexity features, the middle row (c, d) refers to standard features, and the lower row (e, f) to a combination of both.

becomes even more evident in this scenario (Figure 8.3 d). The regions are clearly separable with considerable space between each other. We find an interesting “Beethoven outlier” among J. S. Bach’s pieces (the point lies at about $(92, -71)$). This point belongs to the piece “Trauermarsch für Eleonore Prochaska,” a short piece in B minor with a dotted rhythm, which is characteristic for a funeral march. Though being written for full orchestra, the instrumentation is mainly dominated by the wind instruments. This possibly results in a

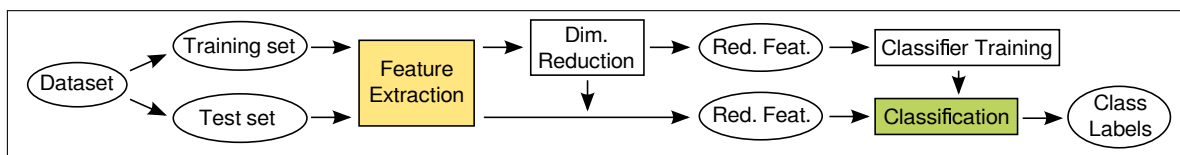


Figure 8.4. Schematic overview of the classification procedure. For applying cross validation, we split the datasets into training and test set. After extracting features for both sets, we perform dimensionality reduction (LDA) to $Z - 1$ dimensions on the training set and transform the test set features with the resulting matrix. On the reduced features, we train and test a machine learning classifier.

unique timbral character, which is—in our dataset—more similar to the timbre of some of the pieces by J. S. Bach. Here, one might argue that a timbral similarity due to the use of certain instruments constitutes a *stylistic* similarity as well. Just as in the five-composer case, the combined features lead to a similar picture than standard features alone. The Beethoven outlier is still visible but slightly better separated from Bach’s pieces. Overall, we see that we have to be careful when interpreting such graphs. Even if a clear separation is possible with some kind of audio features, this does not necessarily constitute a *meaningful* separation based on human-interpretable *musical* reasons.

8.3 Classification Experiments

8.3.1 Classification Procedure

In this section, we present detailed results of our classification experiments. First, we want to describe the experimental design. We employ a standard classification procedure as typically used for MIR experiments (compare Section 3.6). As the first step, we calculate a feature matrix using different configurations of our chroma-based features. Moreover, we test standard audio features as a baseline approach (see Section 8.3.4 for more details on the features’ influence). As discussed in the previous section, we then apply dimensionality reduction (LDA) in order to avoid problems due to the “curse of dimensionality.” We compute the LDA transformation matrix on the basis of the (labeled) training data and apply the resulting transformation to the test set’s features as well. For the output dimensionality L , we always use $L = Z - 1$ with Z denoting the number of classes in the classification problem. This results in $L = 3$ reduced feature dimensions for all *Cross-Era* subsets, $L = 10$ dimensions for *Cross-Comp-11*, and $L = 4$ dimensions for *Cross-Comp-5*. On the LDA-reduced features of the training set, we then train a classifier (Section 8.3.2). After performing classification on the test set, we calculate the fraction of correctly classified test instances and average this number over all classes (mean classification accuracy).

To optimally exploit our available data, we perform cross validation (CV) as presented in Section 3.6.3. For all our experiments, we use $Y = 3$ folds resulting in $2/3$ of the dataset for training and $1/3$ for testing. A higher number of folds may lead to better classification performance. However, the danger of overfitting with respect to semantically meaningless factors simultaneously increases. Furthermore, we want to study the robustness of the classification with respect to the randomized distribution of folds. To this end, we repeat the whole classification procedure ten times with re-initialized folds and average over the results.

Beyond the mean classification accuracy, we calculate deviations with respect to different evaluation steps in order to better estimate the classification performance for unseen data:

- **Inter-Run Deviation.** Here, we compute the standard deviation of the mean multi-class accuracy over the ten runs of the whole experiment. This is a measure for the stability of the results with respect to different fold partitionings since we randomly re-initialize the fold partitioning for every run. For an ideal scenario, we would expect an inter-run deviation of zero since the training should not depend on the data selection. In the opposite case, we may have a high impact of the fold partitioning on the learning success. Then, the classification accuracies obtained in the ten CV runs may considerably differ from each other leading to a high inter-run deviation.
- **Inter-Fold Deviation.** This measure relates to the stability of the results throughout one single cross validation procedure. For each of the three CV rounds (compare Figure 3.18), we use one of the three folds as test data and obtain a mean accuracy. We calculate the standard deviation of the three accuracy values. This measure indicates how much the accuracies for the three test folds differ from each other. If many instances are classified correctly in one fold but much less in the other fold, we obtain a high inter-fold deviation. Finally, we average this standard deviation over all 10 runs (repetitions of the whole CV).
- **Inter-Class Deviation.** For the third measure, we consider the individual class accuracies—the fraction of correctly classified instances for each class. We calculate the standard deviation over these individual class accuracies. This value indicates how balanced the results are between the classes. A bad classifier assigns most of the test instances to one or few classes while other classes do hardly obtain any instance. This leads to high accuracies for the preferred classes at the cost of low accuracies for the others. Therefore, a high inter-class deviation points to a bias towards one or few of the classes. We calculate this measure for every run (CV repetition) and finally average over the ten runs.

8.3.2 Influence of the Classifiers

First, we present classification results obtained with three different classifiers. As features, we use the three configurations from the previous section, namely chroma-based features (interval, chord, and complexity features on the basis of NNLS chroma in four time scales, 136 features in total), standard spectrum-based features (MFCC, OSC, ASE, Loudness and more, 238 dimensions in total), as well as the combination of both. We try out three different classifiers as presented in Section 3.6.3. As an example for a generative classifier, we use a Gaussian Mixture Model (GMM) with $G_{\text{GMM}} = 10$ multivariate Gaussians. Furthermore, we employ a Support Vector Machine (SVM) as implemented in the public LIBSVM library [32] with an RBF kernel and a two-stage grid search for optimizing the kernel parameters C_{SVM} and γ_{SVM} .⁵ Finally, we make use of a Random Forest (RF) classifier in the WEKA implementation [84].

Table 8.3 displays the results of the study. For the *Cross-Era* set and its subsets (blocks 1–3), the performance is high in general ($> 80\%$ accuracy for most scenarios). In comparison, a random guess would obtain 25% on average for four classes. Only the RF classifier obtains slightly worse results of about 70% for *Cross-Era-Piano* and *Cross-Era-Orchestra* on the basis of the combined features. Besides these two outliers, all three classifiers perform similarly for the *Cross-Era* data. In general, the SVM results are slightly higher than the others’.

⁵For the grid search, we run an internal five-fold cross validation on the training set only.

Table 8.3. Classification results for different classifiers and datasets. For five data subsets and three feature configurations, we show the classification results of three different classifiers in a three-fold cross validation over ten runs. Beyond the mean classification accuracy, we display standard deviations with respect to three different parameters. We use LDA transformation to reduce the initial dimensionality of the feature space with respect to the number of classes.

<i>Feature Types</i>	Chroma-based features			Standard features			Combined features		
<i>Dimensionality</i>	136 $\rightarrow L \in \{3, 10, 4\}$			238 $\rightarrow L \in \{3, 10, 4\}$			374 $\rightarrow L \in \{3, 10, 4\}$		
<i>Classifier</i>	GMM	SVM	RF	GMM	SVM	RF	GMM	SVM	RF
<i>Cross-Era-Full</i> ($L = 3$)									
Mean Accuracy	83.4%	84.3%	82.7%	86.6%	87.0%	85.4%	92.1%	92.2%	90.0%
Inter-Run Dev.	0.5%	0.6%	0.7%	0.6%	0.5%	0.7%	0.5%	0.7%	0.7%
Inter-Fold Dev.	1.5%	1.4%	1.4%	1.2%	1.2%	1.1%	0.8%	0.8%	1.4%
Inter-Class Dev.	3.2%	2.3%	3.2%	8.2%	8.2%	8.4%	3.8%	3.4%	4.7%
<i>Cross-Era-Piano</i> ($L = 3$)									
Mean Accuracy	84.0%	86.0%	83.8%	87.3%	88.0%	85.9%	85.5%	86.7%	71.5%
Inter-Run Dev.	1.0%	0.7%	1.0%	1.2%	1.6%	1.1%	1.6%	0.8%	2.4%
Inter-Fold Dev.	1.7%	2.0%	2.4%	1.3%	1.9%	1.7%	2.1%	2.2%	3.4%
Inter-Class Dev.	4.4%	4.1%	4.8%	10.7%	10.0%	11.2%	9.1%	7.8%	13.2%
<i>Cross-Era-Orchestra</i> ($L = 3$)									
Mean Accuracy	85.3%	87.3%	85.1%	84.5%	85.9%	82.4%	80.3%	82.9%	70.8%
Inter-Run Dev.	1.2%	0.7%	0.7%	1.2%	1.2%	1.3%	1.1%	1.3%	2.5%
Inter-Fold Dev.	1.7%	1.7%	1.4%	2.2%	1.2%	2.7%	2.3%	2.0%	2.7%
Inter-Class Dev.	3.9%	2.5%	4.0%	8.4%	7.6%	7.8%	6.0%	4.8%	5.7%
<i>Cross-Comp-11</i> ($L = 10$)									
Mean Accuracy	61.1%	67.3%	9.3%	80.1%	82.3%	9.3%	81.1%	82.7%	9.6%
Inter-Run Dev.	1.8%	1.1%	0.6%	1.2%	1.4%	0.4%	2.5%	4.3%	0.6%
Inter-Fold Dev.	1.4%	2.1%	0.9%	2.1%	2.8%	1.0%	4.1%	4.7%	0.8%
Inter-Class Dev.	12.2%	10.9%	19.6%	8.7%	7.3%	24.0%	7.6%	6.2%	25.1%
<i>Cross-Comp-5</i> ($L = 4$)									
Mean Accuracy	73.6%	77.2%	72.7%	75.2%	78.0%	68.2%	34.6%	42.7%	41.4%
Inter-Run Dev.	1.4%	2.0%	1.2%	5.3%	2.2%	4.0%	3.2%	5.1%	2.9%
Inter-Fold Dev.	3.1%	2.4%	2.7%	4.5%	3.6%	4.5%	5.6%	7.5%	6.7%
Inter-Class Dev.	6.0%	5.7%	7.1%	9.2%	10.0%	7.4%	9.9%	11.2%	10.9%

Comparing the different feature types, we mostly find weak differences. For *Cross-Era-Full* and *Cross-Era-Piano*, standard features lead to slightly better accuracies than chroma-based features. The orchestra scenario behaves differently. Here, chroma-based features outperform standard features or the combination of both. This is an interesting observation since we would expect a more meaningful classification based on timbral characteristics for orchestral music than for piano. Only for the full dataset, the combination of both feature sets leads to further improvement, which is a surprising observation. Possibly, having these very different feature types at hand may enable the classifier to over-adapt to the training data, which may lead to worse generalization. For the full dataset, this over-adaptation might be prevented by the need to model two timbrally different types of pieces simultaneously.

Though the standard features perform similar or better for the *Cross-Era* subsets, we need to be careful with these results. Looking at the inter-class deviation (lowest row in each block), we find considerably higher values for the standard features with all classifiers and

subsets. This points to more imbalanced results between the classes. We will further discuss such type of behavior in the following sections.

Let us now consider the *Cross-Composer* dataset. Here, the results are worse in general. There may be several reasons for this behavior. First, the number of items per class is lower (100) than for the *Cross-Era* data (400 for *Cross-Era-Full*). Along with this, the pieces in *Cross-Era* are stemming from more different sources (albums, artists) than the pieces in *Cross-Composer*. Therefore, the variety of training data better covers the variances within one class for *Cross-Era*. Apart from this, the scenario itself is harder since we have more classes. In particular, the *Cross-Comp-11* scenario requires a very subtle discrimination between stylistically related composers such as Haydn and Mozart.

Having these characteristics in mind, the results may be judged as quite good. For the *Cross-Comp-11* task, our combined features even outperform the best results in the MIREX classical composer identification task (78% in 2011 with MFCC-like features and a Neural Network [85]), which is fairly comparable to our experiment (see Section 8.4 for a detailed discussion). Concerning the different classifiers, we again find best results for the SVM, closely followed by the GMM. Interestingly, the RF classifier fails completely for the *Cross-Comp-11* scenario (below random guess accuracy). In contrast, RF performs similar to the other classifiers for the reduced composer problem *Cross-Comp-5*. We have no explanation why this classifier only fails for certain scenarios. Not using dimensional reduction (LDA) here did not improve this bad result.

Looking at the different feature configurations, the composer identification tasks seem to benefit from the use of standard features. Especially for the eleven composer problem, the difference to the use of chroma-based features is large (up to 19%). Moreover, the inter-class deviation is smaller for standard features. These observations are in accordance to the LDA visualizations (Figure 8.3) where we observed better separation with standard features. For the *Cross-Comp-11* scenario, the combination of chroma-based and standard features leads to further improvement. For the five-composer problem, we find a different behavior. Here, the combination of features leads to clearly worse results—much lower accuracies and higher inter-class deviations—than each feature set alone. Hence, the combination of different feature types seems to cause over-adaptation in the training phase.

Regarding the different evaluation measures, we only find slight deviations both for the inter-run and inter-fold deviations. These measures slightly increase with decreasing classification accuracy. In comparison, the inter-class deviation seems to be more important. As an example, the accuracies for *Cross-Era-Piano* are all quite similar whereas the inter-class deviation considerably changes. For the following sections, we only consider mean accuracy and inter-class deviation as evaluation measures.

In summary, we found only smaller differences between the different classifiers' performance. The SVM classifier always performed best. In comparison, the GMM results came out slightly worse. The RF classifier obtained similar accuracies for most scenarios but failed completely for the *Cross-Comp-11* dataset. Concerning computational complexity and runtime, SVM is by far the slowest method since the grid search optimization is extremely time-consuming. For these reasons, we used the GMM classifier for further experiments. In the following section, we also investigate the influence of the model parameter G_{GMM} (number of Gaussians, see Figure 8.5).

Table 8.4. Classification results with filtering. We display the performance of a GMM classifier with $G_{\text{GMM}} = 10$ Gaussians. For all data subsets and three feature configurations, we compare the results with and without filtering instances for the cross validation. We use dimensionality reduction (LDA) resulting in L -dimensional features as input for the classifier.

<i>Feature Types</i>	Chroma-based features		Standard features		Combined features	
<i>Dimensionality</i>	136 $\rightarrow L \in \{3, 10, 4\}$		238 $\rightarrow L \in \{3, 10, 4\}$		374 $\rightarrow L \in \{3, 10, 4\}$	
<i>Filter</i>	–	Composer	–	Composer	–	Composer
<i>Cross-Era-Full</i> ($L = 3$)						
Mean Accuracy	83.5%	72.7%	86.5%	54.0%	92.1%	67.7%
Inter-Class Dev.	3.4%	6.9%	8.7%	7.5%	3.8%	12.5%
<i>Cross-Era-Piano</i> ($L = 3$)						
Mean Accuracy	84.4%	69.6%	87.6%	35.8%	85.5%	44.2%
Inter-Class Dev.	4.3%	6.6%	10.2%	18.9%	9.1%	22.4%
<i>Cross-Era-Orchestra</i> ($L = 3$)						
Mean Accuracy	85.9%	77.7%	84.6%	70.2%	80.3%	67.7%
Inter-Class Dev.	3.5%	6.9%	7.4%	9.8%	6.0%	7.0%
<i>Filter</i>	–	Artist	–	Artist	–	Artist
<i>Cross-Comp-11</i> ($L = 10$)						
Mean Accuracy	61.5%	37.4%	80.3%	35.7%	81.1%	38.9%
Inter-Class Dev.	13.0%	12.5%	8.2%	22.5%	7.6%	22.0%
<i>Cross-Comp-5</i> ($L = 4$)						
Mean Accuracy	54.9%	54.0%	71.7%	47.5%	34.6%	27.3%
Inter-Class Dev.	10.9%	11.4%	10.9%	28.9%	9.9%	11.0%

8.3.3 Influence of the Cross Validation Design

Inspired by previous MIR research [63, 178], we want to examine our classification procedure with respect to the partitioning of the cross validation folds. As usual in genre classification datasets, the classes in *Cross-Era* and *Cross-Composer* often contain several tracks from one album. These tracks exhibit not only stylistic similarity but may have typical characteristics due to the artists, the recording conditions or audio post production steps. In the CV procedure, this may lead to overfitting due to the so-called “album” or “artist effect.” If both training and test folds contain items from the same CD recording, the system can adapt to technical artifacts or the specific sound of a recording rather than learning musically meaningful properties [63, 178]. Additionally, we want to avoid substantial influence of a specific composer style on the classification but capture the overall style characteristics of a period. Motivated by these considerations, we apply a “composer filter,” which forces a composer’s works to be in the same fold, thus avoiding the album effect and a “composer effect” at the same time.⁶

Ideally, it would be useful to separate album- and composer-filtering. Applying a composer filter makes the classification task considerably harder since the classifier gets no training data from a composer to learn its style. Unfortunately, we do not have album or artist annotations for the *Cross-Era* set. For *Cross-Composer*, we have such annotations regarding the artists.⁷ Therefore, we apply composer filtering for *Cross-Era* and artist filtering for *Cross-Composer*—having in mind that these filters have different effects.

⁶The *Cross-Era* dataset does not contain works by different composers that are on one album.

⁷With the term “artist,” we refer to the interpreter of a piece such as the soloist or the orchestra.

Table 8.4 presents classification results for evaluating the effects of filtering. In general, the use of filtering leads to a considerable decrease in accuracy. Furthermore, the inter-class deviation increases for most configurations. For the standard features, the loss of performance is extreme. Looking at the *Cross-Era-Piano* data, the accuracy using standard features drops from 87.6 % to 35.8 % when using the composer filter, which is already close to the chance level result of 25 %. For the complex scenario *Cross-Comp-11*, the high performance of 80.3 % goes down to 35.7 % only. From such observations, we conclude that classification based on standard features—which mostly capture timbral characteristics—is not sufficient to learn musical styles since a massive overfitting to timbral artifacts may occur. The situation is slightly better for *Cross-Era-Orchestra* where we still achieve 70.2 % accuracy with standard features.

For the chroma-based features, filtering also leads to a decrease in performance but, to a much smaller extent. Classifying *Cross-Era-Full*, we still obtain 72.7 % accuracy compared to 83.5 % without composer filter. The subsets of *Cross-Era* behave similarly. For the *Cross-Comp-11* data, the decrease is more extreme. Here, the accuracy drops from 61.5 % to 37.4 %. Hence, the album effect also affects classification performance when using chroma-based features. However, the *Cross-Composer* dataset is rather small consisting of 100 instances per class, which—in some cases—stem from a small number of different albums. For this reason, CV with album filtering may considerably reduce the variability of the data with respect to musical properties such as key, mode, tempo, or instrumentation. Thus, it would be helpful to conduct these experiments with a larger dataset. For the subset *Cross-Comp-5*, the situation is quite different. Here, we almost obtain the same result of 54 % when using filtering. This indicates that chroma-based features may be useful for capturing style in a musically meaningful way.

For the combination of chroma-based and standard features, we also find decreasing performance when using filters. Interestingly, the combination of features obtains worse results than chroma-based features alone. For *Cross-Era-Full* and *Cross-Era-Piano* with composer filter, the combined features’ accuracies lie between the accuracies of chroma-based and standard features. From this, we might conclude that the better performance of chroma-based features gets affected when combined with standard features. As a possible reason, the inclusion of standard features might lead to a different training behavior that may rely on properties that are not relevant for style. We see that classification with chroma-based features is the most “stable” scenario with respect to the filtering step. Only for *Cross-Comp-11*, the combined features achieve best results with filtering.

In the next experiment, we evaluate the impact of classifier complexity with respect to the different scenarios. For the GMM classifier, we can adjust the model complexity via the number of Gaussians G_{GMM} . A model with many Gaussians can thoroughly adapt to the shape of the training feature space. Figure 8.5 shows classification results over this parameter. Surprisingly, performance decreases with increasing model complexity—we find the best results for the simple case of $G_{\text{GMM}} = 1$. This may be due to the cross validation procedure. A more complex model may tend to over-adaptation towards the training data, which leads to a worse generalization. The *Cross-Composer* dataset (Subfigures b, d) seems to be more sensitive to this effect, which may be due to its smaller size per class. Moreover, composer identification with artist filter exhibits large values for the inter-class deviation. This points to the classification being highly biased towards few composers. Due to these results, we use a GMM with $G_{\text{GMM}} = 1$ in the further experiments, which, additionally, constitutes a computationally fast classifier. The observations in this section may indicate

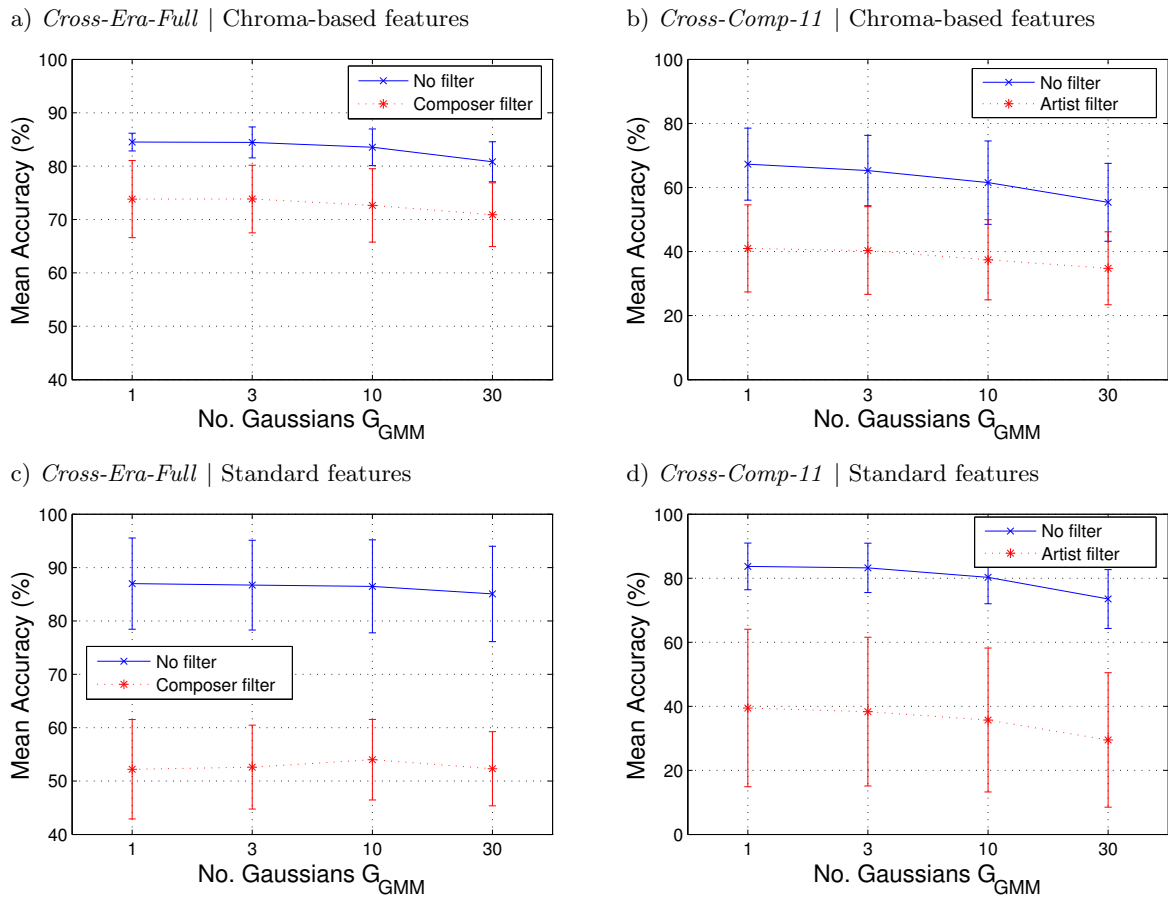


Figure 8.5. Classification results for varying model complexity. For the two datasets *Cross-Era-Full* (left hand side) and *Cross-Comp-11* (right hand side), we plot the classification accuracy of a GMM classifier over the number of Gaussians G_{GMM} . The error bars correspond to the inter-class deviation. The results in the upper row (a, b) rely on chroma-based features, for the lower row (c, d), on standard features. In all cases, we perform LDA to reduce the feature dimensionality to $L = 3$ (*Cross-Era*) and $L = 10$ (*Cross-Comp-11*), respectively. The blue lines indicate the results without filtering. For the red curves, we applied composer filtering (for *Cross-Era*) or artist filtering (for *Cross-Composer*) in the cross validation.

that the chroma-based features capture some “musical” information that is not related to timbre but to tonal aspects.

8.3.4 Influence of the Feature Types

We now want to examine the efficiency of the different feature types in more detail. First, we investigate the influence of the time scale for computing chroma-based classification features (compare Section 6.1.2.2). For this study, we refer to [256] where we presented results for *Cross-Era* with a different setting (ten-fold cross validation, SVM classifier, no LDA reduction, no grid search, no filtering).⁸ In total, we use seven different temporal resolutions of the chroma features as presented in Table 6.1. Based on these representations, we calculate template-based features Ψ (Section 6.1.3) for the six interval categories $\Psi^{\text{IC}1}, \dots, \Psi^{\text{IC}6}$ and

⁸This experimental configuration is considerably different from the one used in our following studies. However, this does not necessarily constitute a problem, since we are not interested in the absolute accuracies but in the *relative* importance of different chroma resolutions for classification.

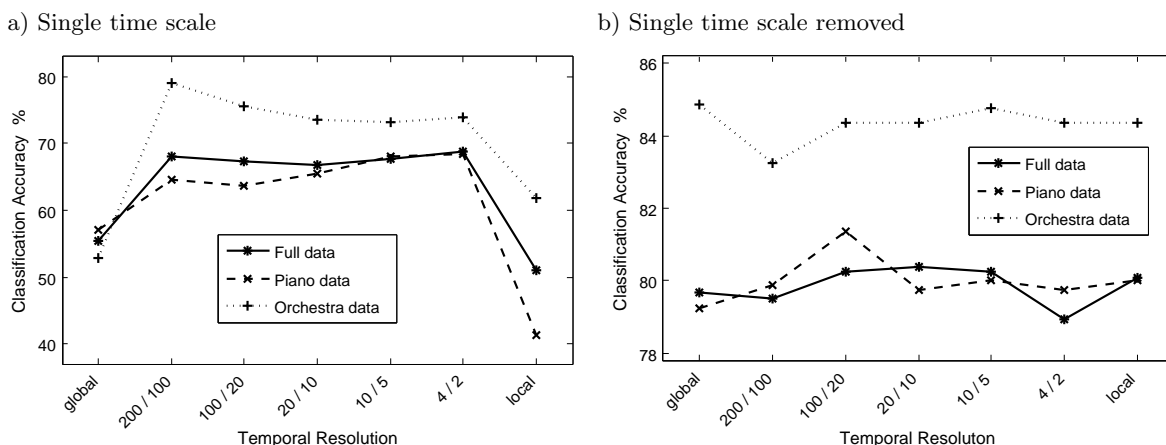


Figure 8.6. Classification accuracy for different temporal resolutions. Based on seven different chromagram resolutions, we derive template-based features for intervals and triads. The left figure (a) shows the accuracy of an SVM classifier using only features based on a single time scale. To the right hand side (b), we used all but one time scale. We obtained these results using ten-fold cross validation.

the four triad types $\Psi^M, \Psi^m, \Psi^\circ, \Psi^+$. Calculating the mean of the local values, we obtain 10 features per time scale for each piece.

Figure 8.6 shows the results of this study. In one test (a), we used only one temporal resolution (10 feature dimensions). In the other scenario (b), we left out the respective time scale ($10 \times (7 - 1) = 60$ dimensions). Here, we do not use dimensionality reduction (LDA). The results confirm our assumption that, for a powerful classification, more than one time scale is necessary. Only relying on the global scale leads to bad results since a 12-dimensional global chroma statistics cannot represent the tonal characteristics of the music in all details. Nonetheless, the local and fine scales alone are not sufficient for a good classification either. Leaving out one of the medium resolutions only slightly affects the performance. For all other experiments, we confine ourselves to use the four different time scales $[\text{Chroma}]_{\text{global}}, [\text{Chroma}]_{100}^{200}, [\text{Chroma}]_5^{10}$, and $[\text{Chroma}]_{\text{local}}$. Thereby, we keep the variety of different resolutions including global and local scale.

For the following studies, we use the GMM classifier with one Gaussian. For better understanding the real-world behavior for unseen data, we use composer filtering for *Cross-Era* and artist filtering for *Cross-Composer*, respectively. Table 8.5 shows the results of a large study regarding different feature types. We averaged all results over 10 runs of the 3-fold CV and give the inter-class deviations (compare Section 8.3.2). In the first block, we display the classification results for the four different chroma implementations Chroma Pitch (CP, [165]), Chroma Log Pitch (CLP, [165]), Enhanced Pitch Class Profiles (EPCP, [131]), and Non-Negative Least Squares chroma (NNLS, [147]). For details of the implementations, we refer to Section 3.5.3. We smoothed the chromagrams to four different temporal resolutions and calculated template-based features for intervals and triads as mentioned above. Furthermore, we calculated the seven types of complexity features (Section 6.2.3). From the local features, we computed the arithmetic mean and the standard deviation in order to obtain piece-level classification features. Therefore, we end up with $2 \times (6 + 4 + 7) \times 4 = 136$ feature dimensions for each chroma implementation before applying LDA reduction to $L = Z - 1$ dimensions.

First, let us discuss the impact of the chroma computation. For the three *Cross-Era* subsets, the NNLS chroma performs best—followed by CP. The enhancement strategies of CLP (logarithmic compression) and EPCP (overtone removal by spectral multiplication) do

Table 8.5. Classification experiments for different feature types. We classify all five subsets with a GMM classifier ($G_{\text{GMM}} = 1$). For both scenarios, we apply filtering (composer filter for *Cross-Era* and artist filter for *Cross-Composer*). “Dim.” indicates the initial number of feature dimensions *before* applying dimensionality reduction. In the “Dev.” column, we display the inter-class deviation. For the chroma-based features (Complexity, Interval, and Triads), we always use the four time scales $[\text{Chroma}]_{\text{global}}$, $[\text{Chroma}]_{100}^{200}$, $[\text{Chroma}]_5^{10}$, and $[\text{Chroma}]_{\text{local}}$.

<i>Dataset</i>		<i>Cross-Era</i>		<i>Cross-Era</i>		<i>Cross-Era</i>		<i>Cross-Comp</i>		<i>Cross-Comp</i>	
<i>Subset</i>		<i>Full</i>		<i>Piano</i>		<i>Orchestra</i>		<i>11 Comp.</i>		<i>5 Comp.</i>	
<i>Reduced dimensionality L</i>		3		3		3		10		4	
<i>Features</i>	<i>Dim.</i>	<i>Acc.</i>	<i>Dev.</i>	<i>Acc.</i>	<i>Dev.</i>	<i>Acc.</i>	<i>Dev.</i>	<i>Acc.</i>	<i>Dev.</i>	<i>Acc.</i>	<i>Dev.</i>
Compare Chroma Feature Types (Complexity + Intervals + Triads)											
CP-based	136	71.6%	4.4%	66.6%	13.3%	77.1%	1.9%	37.1%	12.6%	57.8%	7.5%
CLP-based	136	67.6%	8.2%	58.6%	17.7%	75.5%	2.8%	32.7%	13.7%	54.2%	7.9%
EPCP-based	136	66.9%	7.8%	56.4%	11.8%	76.0%	4.5%	36.0%	14.4%	57.8%	12.9%
NNLS-based	136	73.9%	7.2%	72.7%	7.3%	79.1%	6.4%	40.1%	13.4%	55.8%	11.4%
Compare Secondary Feature Types (NNLS-based)											
Complexity + Intervals + Triads	136	73.9%	7.2%	72.7%	7.3%	79.1%	6.4%	40.1%	13.4%	55.8%	11.4%
Complexity	56	67.1%	7.9%	65.1%	6.7%	74.8%	5.8%	35.8%	13.8%	56.5%	8.4%
Intervals + Triads	80	74.6%	6.9%	73.7%	4.9%	79.4%	6.3%	39.2%	10.1%	57.5%	9.8%
Intervals only	48	70.9%	9.0%	71.2%	5.6%	78.7%	4.5%	37.2%	11.6%	54.6%	9.0%
Triads only	32	70.2%	10.0%	66.3%	8.8%	78.4%	6.1%	38.7%	13.8%	58.4%	7.1%
Influence of Chord Progressions											
Chord progr.	55	65.9%	11.1%	56.1%	13.3%	68.8%	6.6%	28.5%	15.8%	44.0%	20.1%
Chord progr. + NNLS-based	191	73.7%	6.0%	70.3%	6.4%	79.6%	5.7%	42.6%	13.2%	55.9%	5.9%
Chord progr. + Intervals + Triads	135	75.5%	5.2%	72.5%	4.5%	78.8%	6.6%	42.4%	10.9%	58.8%	4.9%
Chord progr. + Complexity	111	70.9%	6.2%	65.7%	5.2%	78.0%	4.4%	41.4%	13.1%	59.6%	5.2%
Combinations with Standard Features											
Standard only	238	52.7%	8.9%	36.3%	20.5%	71.8%	9.2%	38.5%	23.5%	50.0%	32.1%
Standard + NNLS-based	374	67.7%	14.2%	44.6%	22.1%	71.0%	7.0%	42.0%	22.0%	30.6%	11.6%
Stand. + Chord pr.	293	62.7%	12.6%	40.8%	21.5%	74.8%	7.7%	40.6%	21.3%	44.6%	23.8%
Stand. + Chord pr. + NNLS-based	429	67.7%	14.4%	45.6%	21.1%	71.7%	7.2%	41.8%	21.3%	46.4%	22.1%

not seem to be beneficial for deriving classification features. For CLP, this is no surprise since logarithmic compression makes the features less distinct. Regarding instrumentation, the differences are most remarkable for piano data. For *Cross-Comp-11*, the situation is similar to the *Cross-Era* results. In contrast, CP and EPCP features perform better than NNLS in the *Cross-Comp-5* scenario. Due to their overall good performance, we rely on NNLS chroma for deriving classification features in all further experiments.

The next block displays results obtained with different chroma-based features. To this end, we computed template-based features for intervals and triads as well as complexity features from NNLS chromagrams in four temporal resolutions. Comparing complexity features with template-based interval and triad features, the latter ones lead to better accuracies.

Looking at the templates in more detail, both interval templates and triad templates result in a meaningful classification. For *Cross-Era*, we find a slightly better performance when using interval templates. With the *Cross-Composer* scenario, the results are better for triad templates—which even outperform the combination of both for the *Cross-Comp-5* dataset. For all scenarios except for the *Cross-Comp-11* case, template-based features alone perform even better than in combination with complexity features. We may see this as a motivation to test more advanced templates (modeling seventh chords or more dissonant sonorities) in future work.

In the third block, we show classification results using chord progressions as classification features. From the chords obtained with the Chordino algorithm [147], we calculate the relative frequency of general root note progressions (for any chord types) and the root note progressions with respect to the chord types (Major \rightarrow Major, Major \rightarrow Minor, Minor \rightarrow Major, Major \rightarrow Minor). Ignoring the self-transitions, we end up with $11 \times 5 = 55$ feature dimensions (see Section 7.2.2 for more details). Using chord progression features alone already results in remarkable performance (65.9% for *Cross-Era-Full*). In combination with the chroma-based features, the results are better. For *Cross-Era-Orchestra* and *Cross-Comp-11*, this configuration leads to the overall best result. Interestingly, leaving out features does not always lead to worse performance. For *Cross-Era-Full*, the results are better when combining chord progressions with template-based features only (overall best performance). For *Cross-Comp-5*, chord progressions and complexity features together result in the highest accuracy. The reasons are not very clear. Maybe, the classifier obtains better generalization with a smaller initial dimensionality. In summary, chord progression bigrams seem to be beneficial for classifying. We may see this as a motivation to test longer n -grams (with $n > 2$) as well.

Finally, we want to investigate the combination of tonal features with standard features. As we mentioned in the previous section, standard features do not lead to better accuracies—as soon as we use composer or artist filtering. Here, we observe a similar behavior. For *Cross-Era*, standard features alone lead to good results for orchestral data. Since standard features mostly capture timbral properties, this may be due to the individual sound and instrumentation of each style period. The piano case seems to fail with standard features (with 36.3% only about 10% above chance level), which is in accordance with our results in [258]. Beyond worse accuracies, the inter-class deviation is higher for most scenarios including standard features. For the *Cross-Composer* tasks, standard features seem to be more beneficial. As stated above, we assume that this may result from the small size of the dataset in combination with a large variety of instrumentations rather than from “real” stylistic properties. Combining standard features with any kind of tonal feature leads to improvements in most cases. Only for *Cross-Comp-5*, this effect is reversed. Adding NNLS-based template and complexity features leads to a performance decrease of 20%. The reasons for this behavior are not clear. However, the result for standard features alone (50.0%) has a large inter-class deviation and, thus, seems to be highly unbalanced with respect to the individual composers.

Comparing all of these results with respect to the initial feature dimensionality (before performing LDA), we see that a high number of features does not necessarily lead to higher performance. For example, 80-dimensional template-based features lead to the best results for *Cross-Era-Piano*. First, this suggests that our training procedure (dimensionality reduction and cross validation filtering) succeeds in avoiding the “curse of dimensionality.” Furthermore, we suppose that using many features may lead to an over-adaptation to the training data, which results in lower accuracies for the test set and, thus, worse generalization.

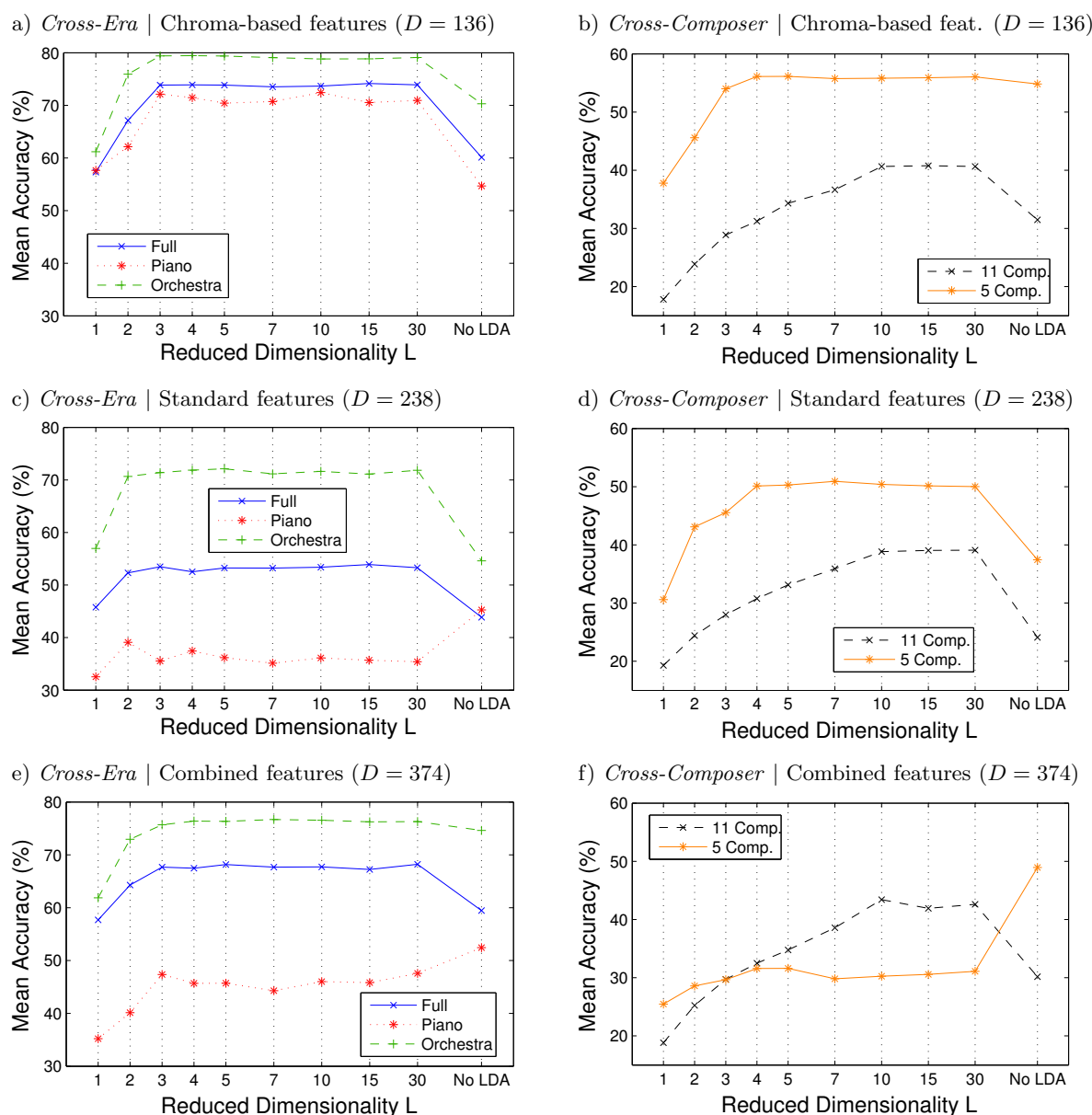


Figure 8.7. Classification results for varying number of LDA dimensions. For testing the dependency of the classification procedure on dimensionality reduction, we conducted experiments with varying number of output dimensions and without using LDA. The left hand side shows the results for the *Cross-Era* subsets, the right hand side for *Cross-Composer*. We performed this study using chroma-based features (interval, chord, and complexity based on NNLS chroma) and standard features as well as their combination. For all experiments, we used a GMM classifier with one Gaussian and composer/artist filtering.

Nevertheless, we observed some unexpected behavior when combining different feature types. Several feature combinations showed considerably worse performance than each feature set alone. Possibly, the dimensionality reduction step (LDA) may influence some of these effects. To test this assumption, we perform further classification experiments with varying output dimensionality L . Figure 8.7 shows the results of this study using a GMM classifier with $G_{\text{GMM}} = 1$ on all five data subsets. Most of the observations confirm our expectations. In general, classification performance steadily increases with the output dimensionality L .

Table 8.6. Classification results of a GMM classifier. For these experiments, we use template-based features for intervals and chords based on four temporal resolutions of the NNLS chroma. We obtained these results by performing 100 CV runs for each dataset using a GMM classifier with one Gaussian and composer/artist filtering.

<i>Dataset</i>	<i>Cross-Era</i>	<i>Cross-Era</i>	<i>Cross-Era</i>	<i>Cross-Comp</i>	<i>Cross-Comp</i>
<i>Subset</i>	<i>Full</i>	<i>Piano</i>	<i>Orchestra</i>	<i>11 Comp.</i>	<i>5 Comp.</i>
<i>Dimensionality</i>	80 → 3	80 → 3	80 → 3	80 → 10	80 → 4
Mean Accuracy	74.60%	73.70%	79.44%	39.28%	57.90%
Inter-Run Dev.	0.85%	1.51%	0.63%	0.95%	0.75%
Inter-Fold Dev.	3.78%	5.63%	3.12%	3.54%	6.22%
Inter-Class Dev.	6.74%	5.50%	6.46%	9.84%	9.22%

At $L = Z - 1$, the curves reach a kind of saturation and do not considerably increase further. For *Cross-Era* ($Z = 4$), we find this point at $L = 3$, for *Cross-Comp-5* at $L = 4$, and for *Cross-Comp-11* at $L = 10$. This behavior is in accordance with our expectations since the LDA transformation only generates $Z - 1$ linearly independent output dimensions. Using no dimensionality reduction at all usually leads to worse performance. For example, the classification accuracy for *Cross-Comp-11* with standard features drops by almost 20% without LDA (Figure 8.7 d). This clearly confirms the “curse of dimensionality.”

Beyond this expected behavior, some scenarios showed very different effects. For *Cross-Era-Piano*, standard features almost always lead to bad performance only slightly above chance level (25%). This accuracy only slightly depends on the LDA dimensionality. Without LDA, we obtain the best result here. This may be an indication that LDA suppresses useful information in that scenario. For the combined features, accuracies are slightly better and behave as expected. However, using no LDA results here in a performance increase as well. We find an even more surprising behavior for the *Cross-Comp-5* scenario. Here, both chroma-based and standard features alone show respectable accuracies and a reasonable behavior. However, the combination of features (Figure 8.7 f) performs much worse and, moreover, increases extremely (by 20%) when not using LDA. We have no explanation for this effect, even though the LDA plots in Figure 8.3 do not show such a behavior. Nevertheless, we may see this as a motivation to test late-fusion approaches, which separately classify using different feature types and then merge the results.

8.3.5 Classification Results in Detail

In the previous sections, we mainly discussed our classification results by looking at the mean classification accuracy and its balance over the classes (*Inter-Class Deviation*). Sturm [233] discussed such type of evaluation in the context of (general) music genre classification. He concludes that only considering mean accuracies may not properly reflect the characteristics of a classification algorithm. To overcome this problem, he suggests to include further “Figures of Merit” into the evaluation—such as confusion matrices or the investigation of constantly misclassified instances. In this section, we want to apply some of these techniques and further show selective results of applying our classification systems to unseen data.

First, let us consider the confusion matrices for some classifiers. We use one of the good performing settings from the previous section, namely a GMM classifier with $G_{\text{GMM}} = 1$, template-based interval and triad features (80 dimensions). The features rely on NNLS chromagrams in four different temporal resolutions. For the experiments of this section, we

a) *Cross-Era-Piano*

Era (correct)	Baroque	66.9	21.5	10.8	0.9
	Classical	15.9	72.4	11.2	0.5
	Romantic	3.2	7.6	78.2	10.9
	Modern	3.8	2.8	16.2	77.3
		Baroque	Classical	Romantic	Modern
		Era (classified)			

b) *Cross-Era-Orchestra*

Era (correct)	Baroque	72.4	20.2	5.6	1.7
	Classical	17.6	75.7	6.8	0.0
	Romantic	6.4	2.9	84.6	6.1
	Modern	0.4	0.0	14.5	85.1
		Baroque	Classical	Romantic	Modern
		Era (classified)			

c) *Cross-Era-Full*

Era (correct)	Baroque	65.2	23.2	10.9	0.6
	Classical	17.0	74.9	8.1	0.0
	Romantic	6.5	5.0	77.7	10.8
	Modern	1.7	0.9	16.8	80.6
		Baroque	Classical	Romantic	Modern
		Era (classified)			

d) *Cross-Comp-5*

Composer (correct)	Bach	51.3	15.7	6.5	13.6	12.9
	Haydn	7.3	63.9	20.4	3.9	4.6
	Beethoven	4.0	20.9	64.3	9.6	1.2
	Brahms	7.9	7.3	8.8	64.9	11.2
	Shostakovich	19.0	2.7	7.4	25.8	45.2
		Bach	Haydn	Beethoven	Brahms	Shostakovich
		Composer (classified)				

e) *Cross-Comp-11*

Composer (correct)	Bach	30.6	14.4	11.6	17.1	0.3	4.3	2.4	11.1	2.7	0.3	5.1
	Handel	11.9	48.8	7.7	5.6	8.7	2.1	1.4	5.4	2.2	3.7	2.6
	Rameau	25.3	14.4	33.2	2.8	4.5	5.6	2.0	4.1	2.5	2.0	3.5
	Haydn	8.8	2.8	0.7	36.9	23.3	12.0	4.3	4.5	1.9	1.2	3.7
	Mozart	4.2	7.3	2.4	24.3	23.4	18.3	1.1	10.4	3.3	1.2	4.1
	Beethoven	2.5	0.3	1.3	12.6	6.4	40.9	3.0	18.4	5.6	6.2	2.8
	Schubert	1.4	3.1	1.2	4.0	2.0	5.0	56.3	6.9	9.1	3.9	7.2
	Mendelssohn	4.3	6.4	3.0	3.3	2.0	11.9	4.7	35.9	17.9	8.1	2.5
	Brahms	2.0	5.3	2.6	0.7	0.6	3.9	8.1	15.2	37.1	21.1	3.4
	Dvorak	2.5	2.9	0.8	0.4	2.1	5.3	2.1	9.8	22.3	49.3	2.8
	Shostakovich	10.3	2.1	4.4	3.4	1.8	5.7	8.4	2.5	8.2	13.4	39.8
		Bach	Handel	Rameau	Haydn	Mozart	Beethoven	Schubert	Mendelssohn	Brahms	Dvorak	Shostakovich
		Composer (classified)										

Figure 8.8. Confusion matrices for the individual datasets. For 100 CV runs with a GMM classifier (including composer/artist filtering), we show the confusion matrices of the classification.

perform 100 runs of the CV with composer and artist filtering, respectively. In Table 8.6, we summarize mean accuracies and three kinds of deviations for this setting. Figure 8.8 shows the corresponding confusion matrices for this experiment. Looking at the *Cross-Era* subsets (a–c), we always find lowest per-class accuracies for the Baroque class. Most frequently, pieces from this class are misclassified as Classical—the “historical neighbor” class—followed by Romantic. Confusions with the Modern class are rare. The next worst accuracy results for the Classical pieces, which the classifier mostly assigns to Baroque and less often, to Romantic. Practically never, instances from the Classical period are confused with Modern. The discrimination between Baroque and Classical seems to be the most difficult task for the classifier.

For all scenarios, Romantic and Modern obtain best results with a slightly better performance on Modern for *Cross-Era-Full* and *Cross-Era-Orchestra*. Most frequently, these classes are confused with each other. Since the evolution of compositional style is a rather continuous process, we expect historically neighboring periods to be stylistically more similar in general than more distant periods such as Baroque and Modern.⁹ For this reason, confusions between these “neighbor classes” may still have some musical meaning. Such errors point to a lack of precision in style classification rather than to a complete fail by overadapting to semantically meaningless characteristics. As we saw in the previous section, a more complex classifier may increase this precision but, on the other hand, often obtains worse generalization. Finally, the “neighbor class” errors may reveal the “ill-definedness” of our four-era classification problem itself (compare Section 2.10).

Let us now discuss the confusions for the *Cross-Composer* datasets (Figure 8.8 d–e). For the five-composer problem, Haydn, Beethoven, and Brahms obtain the best per-class accuracies. Among these, the classifier is mostly confusing Haydn and Beethoven with each other. The Brahms pieces are assigned to Shostakovich most often. In the historical view, these all are “neighbor-class” confusions as mentioned above—with respect to this specific dataset. To better illustrate this, we arranged the classes according to the composers’ lifetime. For Bach, the situation is different. His pieces are mostly misclassified as Haydn (a “neighbor class”) but, closely followed, also as Brahms and Shostakovich. If we try to find a “musical” explanation for these confusions, we might argue that J. S. Bach’s music had great influence on composers of the later periods, in particular. In our Shostakovich data, for example, we included the 24 Preludes and Fugues, which constitute an explicit reference to Bach’s well-tempered piano—not only in the arrangement of movements but also with respect to some musical content. The confusions for Shostakovich confirm such assumptions since 19% of his pieces are classified as “Bach.” However, the “neighbor confusions” with Brahms even exceed this number—leading to a overall bad performance of 45.2% for Shostakovich. For *Cross-Comp-11*, the situation is more complicated. For some composers, we observe considerable “neighbor class errors”—indicated by a darker region around the diagonal. Apart from this, we see some confusions within groups of more than two composers by means of square-like blocks with somewhat darker colors. For example, Bach–Handel–Rameau, Haydn–Mozart–Beethoven, as well as Mendelssohn–Brahms–Dvořák. These confusion structures may point to a homogeneity of style within the groups, which leads to an increase of confusions among the respective composers. For Schubert, we obtain the overall best results of 56.3%. Beyond this, the confusions of Schubert’s pieces are broadly distributed over the other classes, which

⁹Nevertheless, some relationships are in contrast to this argument. At some change points in music history, composer wanted to break with the old style. Later, these styles became popular again and gained influence on the composer. One example is the rediscovery of J. S. Bach’s “St. Matthew Passion” by F. Mendelssohn Bartholdy in 1829 (influence of Baroque style on Romantic composers).

Table 8.7. Examples for consistently misclassified instances. From 100 CV runs, we investigated all instances that obtained a wrong but consistent label over all runs. Here, we display all of these errors that are *not* confusions of “neighbor classes” such as Baroque–Classical or Romantic–Modern. The left column indicates the “true” class. At the very right, we display the automatically determined class label. “Ins.” refers to the instrumentation (P $\hat{=}$ Piano, O $\hat{=}$ Orchestra).

<i>Class</i>	<i>Composer</i>	<i>Piece</i>	<i>Ins.</i>	<i>Classified</i>
Baroque	Bach, J. S.	Well-Tempered Piano 1, Prelude in E \flat minor BWV 853	P	Romantic
Baroque	Bach, J. S.	Well-Tempered Piano 1, Prelude in F major BWV 856	P	Romantic
Baroque	Bach, J. S.	Well-Tempered Piano 1, Prelude in A minor BWV 865	P	Romantic
Baroque	Bach, J. S.	Well-Tempered Piano 1, Prelude in B \flat major BWV 866	P	Romantic
Baroque	Bach, J. S.	Well-Tempered Piano 1, Prelude in B \flat minor BWV 867	P	Romantic
Baroque	Bach, J. S.	English Suite No. 3 in G minor BWV 808, Sarabande	P	Romantic
Baroque	Bach, J. S.	Brandenburg Conc. No. 1 in F major BWV 1046, Adagio	O	Romantic
Baroque	Bach, J. S.	Overture No. 2 in B minor BWV 1067, Badinerie	O	Romantic
Baroque	Bach, J. S.	Overture No. 3 in D major BWV 1068, Gigue	O	Romantic
Baroque	Couperin, F.	27 Ordres, Huitième ordre, IX. Rondeau passacaille	P	Romantic
Baroque	Corelli, A.	Concerto grosso op. 6 No. 2, III. Grave – Andante largo	O	Romantic
Baroque	Lully, J.-B.	Ballet de Xerces LWV 12, Gavotte en rondeau	O	Romantic
Baroque	Purcell, H.	Opera “Dido and Aeneas” Z. 626, Overture	O	Romantic
Baroque	Vivaldi, A.	“The Four Seasons,” RV 293 “Autumn,” Adagio molto	O	Romantic
Romantic	Schumann, R.	Kinderszenen op. 15, “Haschemann”	P	Baroque
Romantic	Grieg, E.	Holberg suite op. 40, Gavotte	O	Baroque
Romantic	Mendelssohn, F.	Symphony No. 4 in A major, IV. Saltarello, presto	O	Baroque
Modern	Shostakovich, D.	Preludes & Fugues op. 87 Fugue No. 1 in C major	P	Baroque
Modern	Shostakovich, D.	Preludes & Fugues op. 87 Fugue No. 5 in D major	P	Baroque

establishes some outstanding position. We observe a similar behavior for Shostakovich, whose pieces are mostly classified as Dvořák (13.4%) and Bach (10.3%). The worst performance occurs for the Mozart pieces (23.4%), which are mostly assigned to Haydn (24.3%). This is the only case where the classifier fails for the majority of instances. Here, one might argue that the stylistic relation between Mozart and Haydn is indeed a very close one. Overall, classification is not very precise. Many confusion pairs obtain values of several percent. However, a closer look into the nature of the confusions reveals some relationships that may originate from the music itself rather than from purely technical or machine learning errors.

We now want to look at some error cases in more detail. Previously, we discussed that the CV procedure may lead to misclassifications due to an inconvenient fold partitioning. To get an insight into the classifier’s behavior, Sturm [232, 233] suggested to investigate those errors that are *consistently and persistently mislabeled* throughout multiple CV runs. Such instances, which obtain the same wrong class label over all runs, constitute errors that are inherent to the classification model. To this end, we look at the results of the 100 CV runs of the GMM classifier (equivalent to Table 8.6) for the *Cross-Era-Full* dataset. In total, we found 25.33% errors on average. From these, 11.06% (177 instances) are consistent and persistent misclassifications. This is quite a high number since it affects 43% of all errors. Looking at the type of misclassification, we found that 158 of them constitute “neighbor class errors” such as Classical–Romantic. As we discussed above, this points to a low precision or “sharpness” of the classification rather than to completely meaningless results. A GMM classifier with one Gaussian and previous LDA reduction may just not be able to properly resolve the borders in the overlap regions with chroma-based features only (compare Figure 8.1 a).

Table 8.8. Era classification for unseen data. For this experiment we trained our GMM classifier with chroma-based features on the whole *Cross-Era-Full* dataset (no cross validation). With the resulting model, we classified the *Cross-Comp-11* dataset. This table shows the number of pieces of each composer that are classified to each of the periods (100 pieces per composer in total). The left part refers to a classification experiment with using LDA, the right part without LDA.

<i>Classified Era</i>	Baroque	Classical	Romantic	Modern	Baroque	Classical	Romantic	Modern
	With LDA				Without LDA			
Bach	5	0	75	20	68	5	9	18
Handel	3	0	64	33	56	23	15	6
Rameau	1	0	77	22	69	22	6	3
Haydn	0	0	92	8	25	53	19	3
Mozart	2	1	87	10	28	51	7	14
Beethoven	0	0	91	9	16	37	38	9
Schubert	0	0	78	22	7	16	24	53
Mendelssohn	0	0	91	9	15	19	55	11
Brahms	0	0	92	8	6	13	69	12
Dvořak	1	0	84	15	14	17	65	4
Shostakovich	0	2	83	15	15	2	8	75
Σ Instances	12	3	914	171	319	258	315	208

Let us now consider the 19 remaining errors—consistently misclassified and no “neighbor classes.” Table 8.7 lists the composers and titles of these pieces. The most frequent case are Baroque pieces classified as Romantic. Among these, most are pieces by J. S. Bach. We find five Preludes from the first book of the “Well-Tempered Clavier” as well as several Suite movements. Some of the errors may be “justified” musically. For example, the movement from the first “Brandenburg Concerto” constitutes a slow and lyric piece in minor key. Vivaldi’s “Autumn” movement is also very atmospheric and broad. Other cases are less clear such as the two overture movements including the famous “Badinerie.” With a fluid and monotonous motion and a typical formal shape, both seem to be rather typical for Baroque suites. For the opposed case (Romantic pieces misclassified as Baroque), we find three examples. From these, the Gavotte from Grieg’s “Holberg suite” indeed reminds of a Baroque suite movement. In contrast, the other two cases are less obvious. Finally, we also find two of the Shostakovich fugues to be consistently mislabeled as Baroque. Here, we should mention that even more (eight) movements from this work cycle were assigned to the Romantic class. Since these constitute “neighbor class” errors, we did not include them in the table.

As the last experimental results, we want to present two studies of applying our classifiers to completely unseen data—without using CV. For the first one, we used the *Cross-Era-Full* dataset in a way that all instances from one subset (*Cross-Era-Piano* or *Cross-Era-Orchestra*) either serve as training data or as test data only. With this experiment, we can test the capability to generalize over different timbral structures. Training on the piano data and evaluating on orchestral pieces, we obtain 65.4% mean classification accuracy. The reversed case results in a similar performance of 63.5%. Both accuracies are far over chance level (25%). Compared to the CV results of Table 8.6, these results are quite encouraging. In relation to the *Cross-Era-Full* CV performance, we only loose about 10% in accuracy. From this, we conclude that a simple classification model combined with our chroma-based features may achieve a classification that is not perfect but robust to timbral variation.

To further test the classifier’s behavior on unseen data, we trained a GMM model for the complete *Cross-Era-Full* dataset (without CV). We then applied the resulting classification system to the *Cross-Comp-11* data, which includes composers that match the periods well but also transitional composers such as Beethoven or Schubert.¹⁰ In Table 8.8, we show the number of resulting class labels for the 100 pieces of each composer. For the experiment, we used the configuration of the previous section (GMM, one Gaussian, template-based features, NNLS chroma). Surprisingly, this configuration fails completely (left part). Almost all of the pieces (914 from 1100) obtain the “Romantic” label. Only three instances were classified as “Classical.” For the Modern class assignments, most pieces stem from Handel (33), which, in our opinion, is not really meaningful. It is not very clear why LDA reduction leads to such an imbalanced and meaningless classification here.

Repeating this experiment without LDA reduction, the situation changes (right hand side of Table 8.8). Now, the assignment of the four classes is much more balanced. For Bach, Handel, and Rameau, most instances obtained the “correct” Baroque label. Haydn’s and Mozart’s pieces are categorized as Classical mostly. Interestingly, the assignment of Beethoven’s pieces seems to be balanced equally over Classical and Romantic. In contrast, Schubert’s pieces are mainly classified as modern, which is rather surprising. Possibly, the inclusion of singing voice recordings (about 50% of the Schubert examples) leads to this confusion, since singing voice examples were not included in the training dataset *Cross-Era-Full*. Mendelssohn, Brahms, and Dvořak are preferably classified as Romantic, and 75% of Shostakovich’s pieces obtain the Modern label. Ignoring Beethoven and Schubert as “transitional” and taking the pieces of all other composers as “correct” for the aforementioned eras, we obtain an accuracy of 62.3%, which is very similar to the results of the cross-instrumentation study. Though not being very sharp in “stylistic resolution,” our classification system seems to produce some musically meaningful style predictions for the majority of the unseen recordings.

8.4 Discussion

In this chapter, we tested several machine learning algorithms for classifying musical style. For this, we considered two scenarios with respective datasets. To classify pieces according to rather coarse historical periods, we compiled the *Cross-Era* dataset, which contains an equal amount of piano and orchestra pieces for each class and, thus, enables to study the timbre-invariance of such methods. Second, we tested our features for the task of composer identification. To this end, we compiled the *Cross-Composer* dataset. This corpus is fairly comparable to the dataset of the corresponding MIREX task (11 composers) but only includes the limited number of 100 instances per class, which, additionally, may not be perfectly representative for the whole stylistic range of a composer’s oeuvre. Both datasets contain multiple tracks from the same albums in each class. To consider this effect for classification, we used a composer filter (*Cross-Era*) and an artist filter (*Cross-Composer*), respectively. Our goal was to test different kinds of chroma-based features—as introduced in the previous chapters—for the two classification scenarios. As baseline, we compared the results of chroma-based features with standard audio features that mainly rely on spectral properties and describe the timbre of the music.

¹⁰For the vast majority of pieces, these datasets have no overlap. However, single pieces may occur in different interpretations such as Shostakovich’s Preludes, which are present in *Cross-Era* (Ashkenazy) and *Cross-Composer* (Sherbakov). We assume that this does not considerably influence the overall results.

First, we presented some visualizations using a supervised dimensionality reduction technique (Linear Discriminant Analysis). We compared these plots for the use of different features. All configurations were able to roughly separate the periods in *Cross-Era*. However, chroma-based features had problems with resolving Baroque and Classical. In contrast, they obtained good separation of Modern and the rest of the pieces. Standard features could better resolve Baroque and Classical but led to a high overlap between Romantic and Modern. For the *Cross-Composer* data, standard features seem to be more beneficial. Considering the results of the classification experiments, we doubt that this separation is based on *musical* properties of style. Possibly, confounding structures in the spectral domain can be used by the LDA algorithm to separate the classes. In future work, this should be analyzed in more detail.

Using LDA to avoid the “curse of dimensionality,” we performed several classification experiments. We first compared different types of generative (GMM) and discriminative (SVM, RF) classifiers. When using no filtering for the cross validation, the results are very similar for all classifiers. Only the Random Forest classifier seems to fail for the more complex scenarios such as the classification of eleven composers. The reasons for this behavior are not very clear since this classifier obtained good results in other scenarios. Due to its computational efficiency, we used the GMM classifier for all other experiments. Considering different filters for avoiding album- or artist-specific effects, we observed worse results. As we mentioned in [258], this effect was drastical when using standard features only. On the *Cross-Era-Piano* subset, this led to results only slightly above chance level and, thus, a meaningless classification. Chroma-based features came out much less sensitive to such filtering. They even outperformed the combination of standard and chroma-based features. We conclude that standard features mostly capture non-meaningful properties for style and, thus, including standard features leads to an overadaptation in the training phase. Investigating the classifier complexity (for GMM, the number of Gaussians), we found that a very simple model of only one Gaussian leads to the best and most stable results with respect to the album effect. In summary, we can reach classification accuracies up to 90% with a complex classifier (SVM) and standard features. However, these results seem to be highly affected by overtraining to semantically meaningless properties—especially for piano music. In contrast, a simple classifier with chroma-based features may perform considerably worse but is much more robust. Under real conditions, such a systems may constitute a less precise but stable and reliable classifier.

Regarding the feature types, we found only slight differences for the various types of chroma-based features proposed in this work. The template-based features for intervals and chords performed best for the *Cross-Era* cases. For classifying composers, the use of chord progressions turned out useful. Regarding the chroma feature extraction, NNLS chroma seems to be beneficial for most scenarios. Combining chroma-based features with standard features did not improve classification performance in any scenario when using album or composer filtering in the CV. In other work, further types of chroma-based features were tested for classifying the *Cross-Era* dataset. In this context, Schaab [211, 259] performed several experiments to directly use global chroma histograms as classification features combined with automatic key detection. In [259], we discussed the impact of the key detection performance on the classification results in detail. Gräfe [80] extended these experiments to the use of local keys (duration and transition histograms) for classification. In all these publications [80, 211, 259], the proposed features did not lead to an increase of performance for *Cross-Era*. However, the experiments were performed without using CV filtering, which should be done in future work.

In all experiments, the orchestral data could be classified better than the piano or the combined data. We suggest two explanations for this. First, style characteristics may be more pronounced for orchestral music. This could arise from the fact that orchestral music was often dedicated to a larger audience and, thus, may be less complex and outstanding than piano music, or chamber music in general. Second, our chroma-based features could still contain some timbral information, which may be more useful for classifying a purely orchestral data set.

Finally, we discussed our classification results in more detail by looking at confusion matrices and consistently mislabeled items. For this, we used the simple GMM classifier on the basis of template-based features, which yielded one of the best results. From these analyses, we obtained a good intuition for the behavior of our classifier. Though not generating very high accuracies, we could find some musical explanations for several types of confusions and mis-classifications. Indeed, most of the confusions occurred between “historically neighboring” classes. This is encouraging since we assume that such neighbor instances may still exhibit some kind of stylistic similarity. Altogether, the overall high number of confusions may not only point to the deficiencies of our system but may also reflect the ambiguity of the style categorization itself. As discussed in Section 2.10, musical style may be heavily overlaid by the individuality of the single piece.

Comparing our composer identification results with the state-of-the-art systems of the MIREX task [85], we obtain clearly worse performance. Since we do not know the exact composition of the MIREX dataset, we cannot guarantee that our *Cross-Comp-11* dataset is comparable. Even though the MIREX evaluation makes use of an artist filtering step, our scenario may be more ambitious for machine learning algorithms since we have only a small number of instances per class (100). Furthermore, these instances stem from a small amount of albums (CD compilations), which leads to even more unbalanced training scenario when using the artist filter. For this reason, we assume that a larger and more balanced composer dataset would be necessary to realistically compare our algorithms to the MIREX results. Since many of the MIREX submissions show some similarity to our baseline experiments relying on standard features, we would expect these systems to produce considerably lower accuracies in a scenario like our *Cross-Composer* classification.

Overall, we saw that we have to be very careful with the interpretation of classification results. Machine Learning systems may heavily rely on confounding factors such as recording artifacts or artist-specific timbral properties, which results in a bad generalization for real world scenarios. We showed that using tonality-related features based on a suitable chroma implementation may lead to more robust classification systems—even if the cross validation accuracies are lower in some scenarios.

9 Conclusions

In this thesis, we approached the computational analysis of classical music audio recordings with respect to tonality and style characteristics. For this purpose, we proposed novel types of tonal audio features that build the basis for different analysis systems. In particular, we used these features for clustering and classifying audio recordings with respect to style categories. In our classification experiments, we compared the features' performance against a baseline method using standard spectrum-based features. We further tested, to which extent our methods are invariant to variations in timbre and instrumentation.

In general, the automatic analysis of audio recordings with respect to tonal characteristics constitutes a challenging task. For many music scenarios, state-of-the-art systems for music transcription do not yield satisfying results. Therefore, we cannot use automated methods to simply convert audio recordings into symbolic scores, which musicologists usually take as basis for their analysis. For this reason, we consider tonal mid-level representations of the audio data. More specifically, we use chroma features, which serve to locally capture the pitch class content of the music. We discussed and tested several state-of-the-art methods for chroma extraction and showed that they are, to a certain extent, robust to timbral variations. On the basis of such chroma representations, we proposed techniques for measuring the presence of several types of tonal structures. These analysis methods are inspired by music theory. Hereby, we particularly considered such concepts that can be modeled on the pitch class level and, thus, allow for a realization using chroma features.

As one contribution of this thesis, we proposed a novel method to estimate the global key of a musical piece from an audio recording. This method exploits the particular role of the final chord in classical music for estimating the tonic note. Additionally, we performed an analysis of the full piece's predominant diatonic scale in order to decide on the mode. With optimized parameters, this system reached a key detection accuracy of up to 94% on three datasets comprising 478 pieces. We compared our results to a state-of-the-art algorithm [239], which makes use of learning strategies for deriving pitch class profiles. This algorithm reached 98% on a dataset of piano recordings, which we considered in our evaluation as well. With our reimplementations, we could not reproduce this result—probably, due to a different chroma extraction method. On an unseen dataset of 1200 pieces, our version of this baseline algorithm obtained an accuracy of 87.1%. Our proposed algorithm performed slightly worse (85.4%) but still outperformed other approaches.

Furthermore, we extended our global key estimation method to a local approach. We focused on a twelve-key problem by only considering diatonic scales. Similar tasks were previously approached in the field of Music Information Retrieval. For visualizing diatonic scales over time, we used a chroma smoothing procedure followed by multiplicative scale matching. Inspired by music theory, we arranged these visualizations according to the circle of fifths obtaining a spatial arrangement of “diatonic levels.” For several music examples, we showed that our visualization technique can be useful for analyzing modulations and structural aspects of tonality. We further extended the method to cope for non-diatonic scale types and applied this analysis to several audio examples. The presence of different scale types could be observed from the plots. With these scale estimation methods, the

analysis results turned out to sensitively depend on the windowing parameters, which need to be manually adapted. In related publications, our scale estimation method showed success for key segmentation in pop songs [253] and for deriving features for style classification [80].

As a further contribution of this thesis, we presented novel types of chroma-based features that model tonal characteristics of a piece independently of the key. We computed these features on the basis of different temporal resolutions of the chromagram in order to measure tonal properties on several time scales. One type of features proposed in this dissertation serves to quantify the occurrence of interval and chord types. Since chroma features only incorporate pitch class information and no octave labels, these features cannot discriminate between an interval and its complementary, or between inversions of a chord. In Chapter 8, we showed that these features are useful for style classification. Furthermore, we proposed features to quantify the more abstract notion of tonal complexity. Inspired by several musicological hypotheses, we implemented mathematical realizations for chroma-based complexity measures. We analyzed these features' behavior for individual chords and for the head movements of L. van Beethoven's piano sonatas. For the sonatas, our analyses indicated higher complexity in the development phases and a greater individuality of the late sonatas. In Chapter 7, we analyzed complexity features over the history. As one result, we could measure an increase of tonal complexity over the centuries. During the 19th century, global complexity (relating to full movements) increased, in contrast to local complexity (relating to chords). Our classification experiments revealed that complexity features are useful for obtaining robust style classification.

Beyond this, we performed studies to analyze musical styles with unsupervised methods. For this application, we consider both interval type and tonal complexity features together with chord progression bigrams obtained with a public chord detection algorithm [147]. We compiled a balanced dataset comprising 2000 recordings of orchestra and piano music, respectively. Since we had no annotations of composition years, we mapped the feature values for the pieces onto a historical time axis using the composers' lifetime. With this strategy, we visualized feature values for chord progressions and tonal complexity. We could observe an increase of tonal complexity over the centuries. Furthermore, we confirmed the expectation of high complexity values for atonal pieces by composers such as A. Schönberg or A. Webern. Investigating chord progressions, we observed an increase of third relations between the root notes of consecutive chords during the 19th century. Moreover, the ratio of authentic ("falling") and plagal ("rising") progressions seems to be a suitable measure to discriminate between the Baroque (higher ratio) and the Classical style.

Using such features for clustering pieces individually, across composers, and across composition years led to interesting results. Mapping pieces to composition years and then clustering the year-wise averaged features yielded a clustering result in accordance with the traditionally defined eras in music history. Important boundaries between the clusters occurred roughly at the years 1750, 1825, and 1900. In contrast, the reversed procedure—clustering individual pieces first and then mapping the resulting cluster assignments to the years—produced a different picture by showing only very coarse trends. Pieces seem to exhibit higher individuality than the rather "clean" clustering results for the years might indicate. In contrast, averaging the feature values over all pieces by a composer led to a clearer result. Here, most of the composers were assigned to the "correct" cluster according to their lifetime—with two parallel modern clusters. We conclude that looking at a certain variety of works by a composer may be more suitable for analyzing his or her style rather than investigating a single piece. In general, we have to be careful with these results since the long processing chain may be subject to artifacts and propagation of errors. Furthermore, the chord detection algorithm

itself is error-prone. Nevertheless, these errors seem to be less problematic when analyzing large databases. In such a large-scale scenarios, we could observe interesting tendencies that appear to be musically meaningful, to some degree.

As one of our main contributions, we tested the proposed features for classifying audio recordings of Western classical music. For comparison, we used standardized spectrum-based features as a baseline system. We considered two scenarios of subgenres, namely four historical periods (Baroque, Classical, Romantic, Modern) and classical composers (five and eleven composers, respectively). For this purpose, we compiled two datasets. To test the separation of classes in the feature space, we visualized the datasets using dimensionality reduction (Linear Discriminant Analysis) for chroma-based features, standard features, and the combination of both. The plots revealed that chroma-based features have problems to discriminate Baroque and Classical music whereas standard features struggle with discriminating Romantic and Modern music. Possibly, this may point to similar tonal characteristics of Baroque and Classical music and similar timbral characteristics of Romantic and Modern. The combination of both feature types led to a good separation of instances, in general. Moreover, we tested common machine learning classifiers on our datasets in a cross-validation scenario. Here, we found a different situation. Both chroma-based and standard features led to high mean accuracies up to 90 % using different types of classifiers. However, performing classification in a more realistic scenario by applying filtering techniques¹ in the cross validation resulted in a severe deterioration of results. This observation indicates that, without filtering, our system may learn non-meaningful characteristics such as artist-specific properties—known as “album effect” [63, 178]. With filtering, a classification with standard features led to very low accuracies. In contrast, chroma-based features seem to be less prone to overfitting because of the album effect (73 % accuracy for four eras). Surprisingly, the combination of both feature types performed worse than using chroma features alone. Adding standard features seems to negatively affect the robustness of tonal features. Concerning the classifier complexity, a rather simple model (Gaussian Mixture Model with one Gaussian) seemed to result in a robust system when applying filters in the cross validation. Such a model also produced meaningful classification results of up to 62 % for unseen data without using cross validation. Among the tonal features, template-based interval and chord features alone already resulted in good performances. Combining these features with chord progressions and complexity features led to an increase of accuracy in several scenarios. Regarding the different chroma feature types, NNLS chroma features [147] led to best results for deriving tonal features. We also showed that classification with tonal features is timbre-invariant to a certain extent. Training on piano data and evaluating on orchestral data resulted in 65 % accuracy for classifying into four eras.

As a general trend, we observed that classification accuracies are higher for orchestral pieces than for piano pieces. We assume that the instrumentation provides some meaningful stylistic details and even influences chroma-based features—though they are nearly invariant to timbral differences. We now reconsider the quantitative results in one specific setting—a GMM classifier with one Gaussian using interval and triad features based on NNLS chroma. Here, we obtained an accuracy of 75 % for the full dataset. On piano data, the results were similar whereas classifying orchestral data led to better performance (79 %). Comparing these results to existing work is difficult since there are no studies considering the same categories as in our setting (four historical periods) among related publications. For composer classification, our system obtained worse accuracies. Classifying eleven composers resulted

¹We used a composer filter for classifying eras and an artist filter for classifying composers.

in 39% accuracy. For five composers, we obtained an accuracy of 58%. Some researchers used related categories but in conjunction with subclasses of very different genres [108, 189]. Concerning composer identification, authors reported results of 76% for nine composers [98] and 78% for eleven composers in the 2011 MIREX task [85]. In comparison, our system performed clearly worse (39% for eleven composers using tonal features). Considering the fact that our baseline system with standard features produced accuracies of over 80%—which dropped to 36% when using artist filtering—, we doubt that the experimental conditions (datasets and cross validation settings) are comparable to our restrictive filtering procedure. We suppose that a larger dataset with a higher variety of pieces, instruments, and performers within each class could lead to better results.

In summary, this thesis showed that chroma-based analysis of audio recordings may provide meaningful insights into the tonal and stylistic properties of musical pieces. We presented novel methods for analyzing key and scale structures, for measuring the presence of interval and chord types, and for quantifying some kind of tonal complexity in music recordings. In several case studies, we showed that these analyses have the potential to highlight musically meaningful structures. One challenge is that the feature representations sometimes emphasize details that are not relevant for a musical task. These artifacts may propagate and cause misleading analysis results. Furthermore, the underlying musicological models may not be suitable for a particular piece of music. Additionally, a specific recording may exhibit performance-related artifacts. Due to these reasons, it is often not clear if the automated analysis captures some relevant information about the musical work itself or an artifact of the specific representation. Therefore, fine-grained analysis results based on an individual piece or recording have to be taken with care. In particular, such analyses cannot compete with a detailed and reflected analysis by a musical expert, who can also provide an interpretation of the results. However, the manual generation of analyses becomes very time-consuming for larger corpora of musical works. In such scenarios, automated methods unfold their potential since they allow for analyzing a huge amount of pieces with quantitative and objective methods. When analyzing large databases, artifacts of specific pieces may be averaged out and tendencies become visible. The visualizations presented in this thesis confirmed this assumption by showing interesting trends over the course of music history. As another goal of this thesis, we tested the efficiency of our tonal descriptors for clustering and classifying music recordings according to style categories. Though we primarily found a great individuality of pieces, unsupervised clustering of composers highlighted some stylistic similarities that are undoubtedly recognized among musicologists. Classifying pieces according to historical periods resulted in a good performance despite the vagueness of the task and the complexity of the data. For such experiments, it is very important to carefully compile the datasets.

Our experimental results let us conclude that an interdisciplinary collaboration between musicology and audio signal processing can be very promising. In the specific case of an individual piece, the domain-knowledge of musicologists is necessary for adequately conducting and interpreting musical analyses. With appropriate algorithms, such analyses can then be transferred from individual pieces to large corpora that comprise representative parts of the Western classical music repertoire. This strategy allows for quantitatively testing and verifying hypotheses as well as for highlighting far-reaching trends and, thus, may have the potential to open up a new dimension for musicological research.

Appendix

Additional Material

Table A.1. Dictionary file for the Chordino algorithm. This is the “chord.dict” file for configuring the Chordino Vamp plugin. We used this configuration to estimate the chords for the analyses presented in Chapter 7. The first twelve entries refer to the bass notes, which we did not use. The last twelve entries indicate the active pitch classes for the respective chord type. We have considered the four basic triad types as well as five types of seventh chords. Regarding the nomenclature, the part after the first underscore relates to the quality of the basic triad (major, minor, diminished, or augmented). For the seventh chords, we indicate the quality of the seventh interval over the root note after the second underscore. The algorithm automatically generates circularly shifted versions of these templates to account for all twelve possible root notes.

```

_maj = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,1,0,0,0,0
_min = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,0,1,0,0,0,0
_dim = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,0,0,0
_aug = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0
_dim_dim7 = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,1,0,0
_dim_min7 = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,0,1,0
_maj_min7 = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,1,0,0,1,0
_maj_maj7 = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,1,0,0,0,1
_min_min7 = 0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,0,1,0,0,1,0

```

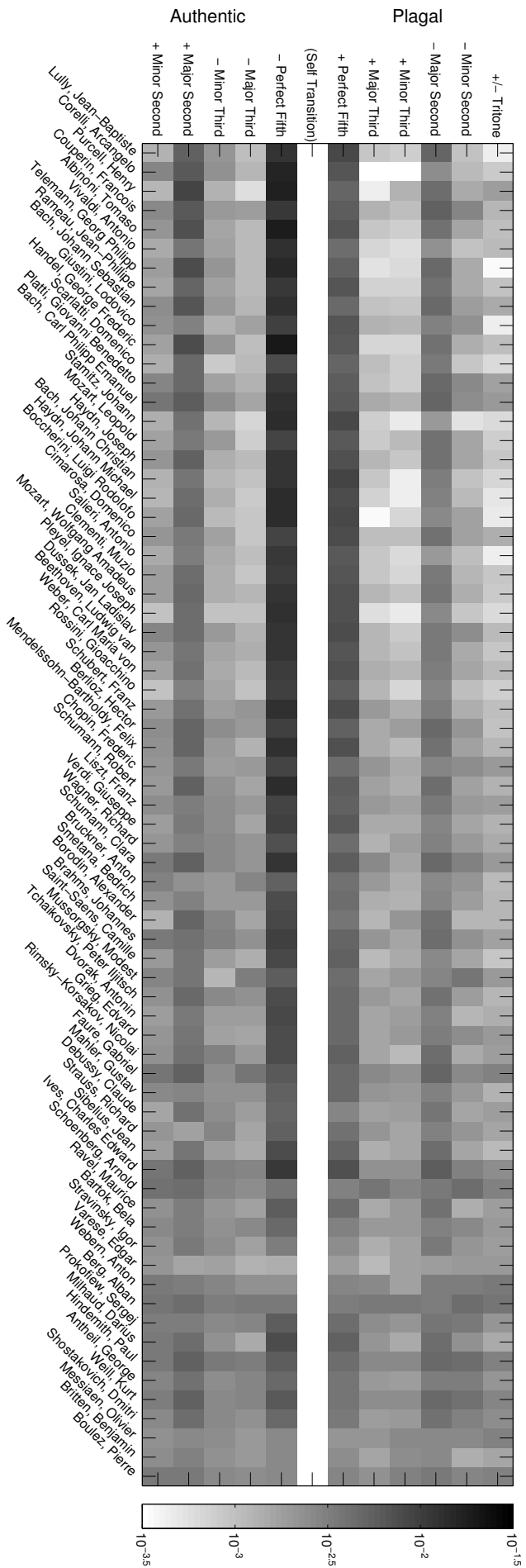


Figure A.1. Root note progressions for the individual composers. Corresponding to Figure 7.5, we show the average frequency of the root note progressions averaged over the individual composers' work. We have arranged the progressions according to plagal ("ascending") and authentic ("descending") progressions. The horizontal axis shows the composers with ascending mean lifetime.

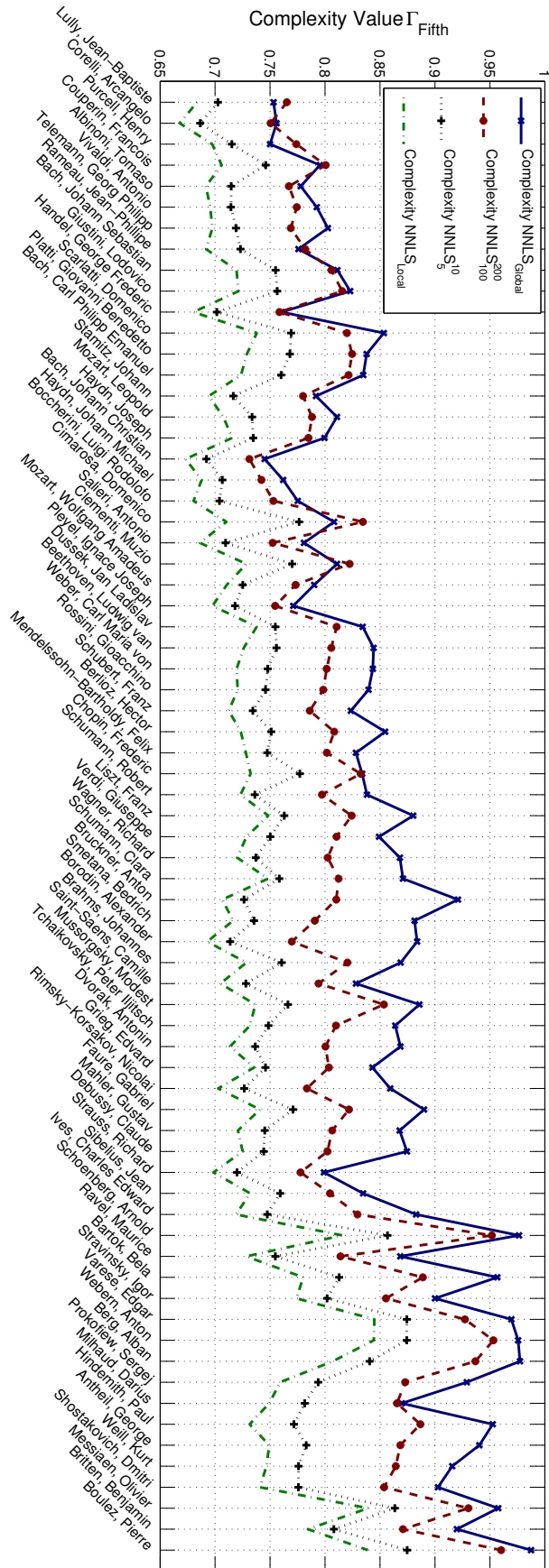


Figure A.2. Average tonal complexity values for individual composers. Here, we show the average tonal complexity for the individual composers' works according to Figure 7.11. The complexity features relate to four different temporal resolutions.

Bibliography

- [1] Toshihiko Abe and Masaaki Honda, “Sinusoidal Model Based on Instantaneous Frequency Attractors,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1292–1300, 2006.
- [2] Toshihiko Abe, Takao Kobayashi, and Satoshi Imai, “Harmonics Tracking and Pitch Extraction Based on Instantaneous Frequency,” in *Proceedings of the 1995 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, 1995, pp. 756–759.
- [3] Guido Adler and W. Oliver Strunk, “Style-Criticism,” *Musical Quarterly*, vol. 20, pp. 172–176, 1934.
- [4] Mark A. Aizerman, Emmanuel M. Braverman, and Lev I. Rozonoér, “Theoretical Foundations of the Potential Function Method in Pattern Recognition Learning,” *Automation and Remote Control*, vol. 25, pp. 821–837, 1964.
- [5] Ethem Alpaydin, *Introduction to Machine Learning*, MIT Press, Cambridge, Massachusetts, 2nd edition, 2010.
- [6] Yali Amit and Donald Geman, “Shape Quantization and Recognition With Randomized Trees,” *Neural Computation*, vol. 9, no. 7, pp. 1545–1588, 1997.
- [7] Chris Anderson, *The Long Tail: Why the Future of Business is Selling Less of More*, Hyperion, New York, 2006.
- [8] Amélie Anglade, Emmanouil Benetos, Matthias Mauch, and Simon Dixon, “Improving Music Genre Classification Using Automatically Induced Harmony Rules,” *Journal of New Music Research*, vol. 39, pp. 349–361, 2010.
- [9] Amélie Anglade and Simon Dixon, “Characterisation of Harmony with Inductive Logic Programming,” in *Proceedings of the 9th International Society for Music Information Retrieval Conference (ISMIR)*, 2008, pp. 63–68.
- [10] Jacques Attali, *Noise: The Political Economy of Music*, vol. 16, Manchester University Press, 1985.
- [11] Wolfgang Auhagen, *Studien zur Tonartencharakteristik in theoretischen Schriften und Kompositionen vom späten 17. bis zum Beginn des 20. Jahrhunderts*, Europäische Hochschulschriften, Reihe 36, Musikwissenschaft, vol. 6, 1983.
- [12] Eric Backer and Peter van Kranenburg, “On Musical Stylometry: A Pattern Recognition Approach,” *Pattern Recognition Letters*, vol. 26, no. 3, pp. 299–309, 2005.
- [13] Gerald J. Balzano, “The Group-Theoretic Description of 12-fold and Microtonal Pitch Systems,” *Computer Music Journal*, vol. 4, no. 4, pp. 66–84, 1980.
- [14] Lajos Bárdos, *Modális Harmóniák (Modal Harmonies)*, Ed. Zenemukiadó, Budapest, 1961.
- [15] Jérôme Barthélemy, “Figured Bass and Tonality Recognition,” in *Proceedings of the 2nd International Symposium on Music Information Retrieval (ISMIR)*, 2001, pp. 129–136.
- [16] Mathieu Barthet, Mark D. Plumbley, Alexander Kachkaev, Jason Dykes, Daniel Wolff, and Tillman Weyde, “Big Chord Data Extraction and Mining,” in *Proceedings of the 9th Conference on Interdisciplinary Musicology (CIM)*, 2014, pp. 174–179.
- [17] Mark A. Bartsch and Gregory H. Wakefield, “To Catch a Chorus: Using Chroma-Based Representations for Audio Thumbnailing,” in *Proceedings Workshop on Applications of Signal Processing (WASPAA)*, 2001, pp. 15–18.
- [18] Mark A. Bartsch and Gregory H. Wakefield, “Audio Thumbnailing of Popular Music Using Chroma-Based Representations,” *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 96–104, 2005.
- [19] Victor Belaiev, “The Signs of Style in Music,” *The Musical Quarterly*, vol. 16, no. 3, pp. 366–377, 1930.
- [20] Richard Ernest Bellman, *Adaptive Control Processes: A Guided Tour*, vol. 4, Princeton University Press, Princeton, 1961.
- [21] Juan Pablo Bello and Jeremy Pickens, “A Robust Mid-Level Representation for Harmonic Content in Music Signals,” in *Proceedings of the 6th International Society for Music Information Retrieval Conference (ISMIR)*, 2005, pp. 304–311.

- [22] David W. Bernstein, "Nineteenth-Century Harmonic Theory: The Austro-German Legacy," in *The Cambridge History of Western Music Theory*, pp. 778–811. Cambridge University Press, Cambridge, 2002.
- [23] Jordi Bonada, "Automatic Technique in Frequency Domain for Near-Lossless Time-Scale Modification of Audio," in *Proceedings of the International Computer Music Conference (ICMC)*, 2000, pp. 396–399.
- [24] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik, "A Training Algorithm for Optimal Margin Classifiers," in *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, 1992, pp. 144–152.
- [25] Leo Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [26] John S. Bridle and Michael D. Brown, "An Experimental Automatic Word Recognition System," in *Joint Speech Research Unit Report*, vol. 1003. Ruislip, England, 1974.
- [27] Judith C. Brown, "Calculation of a Constant Q Spectral Transform," *Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991.
- [28] Manfred F. Bukofzer, *Music in the Baroque Era: from Monteverdi to Bach*, Norton History of Music. W. W. Norton, New York, 1947.
- [29] John A. Burgoyne, Laurent Pugin, Corey Kereliuk, and Ichiro Fujinaga, "A Cross-Validated Study of Modelling Strategies for Automatic Chord Recognition in Audio," in *Proceedings of the 8th International Society for Music Information Retrieval Conference (ISMIR)*, 2007, pp. 251–254.
- [30] Chris Cannam, Christian Landone, and Mark Sandler, "Sonic Visualiser: An Open Source Application for Viewing, Analysing, and Annotating Music Audio Files," in *Proceedings of the ACM Multimedia 2010 International Conference*, Firenze, Italy, 2010, pp. 1467–1468.
- [31] Wei Chai and Barry Vercoe, "Detection of Key Change in Classical Piano Music," in *Proceedings of the 6th International Society for Music Information Retrieval Conference (ISMIR)*, 2005, pp. 468–474.
- [32] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM: A Library for Support Vector Machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1–27, 2011.
- [33] Heng-Tze Cheng, Yi-Hsuan Yang, Yu-Ching Lin, I-Bin Liao, and Homer H. Chen, "Automatic Chord Recognition for Music Classification and Retrieval," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2008, pp. 1505–1508.
- [34] Elaine Chew, *Towards a Mathematical Model of Tonality*, PhD Thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2000.
- [35] Elaine Chew, *Mathematical and Computational Modeling of Tonality: Theory and Applications*, International Series in Operations Research & Management Science. Springer US, 2014.
- [36] Taemin Cho and Juan Pablo Bello, "On the Relative Importance of Individual Components of Chord Recognition Systems," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 2, pp. 477–492, 2014.
- [37] Wei-Ta Chu, Wen-Huang Cheng, and Ja-Ling Wu, "Generative and Discriminative Modeling toward Semantic Context Detection in Audio Tracks," in *Proceedings of the 11th IEEE International Multimedia Modelling Conference*, 2005, pp. 38–45.
- [38] Ching-Hua Chuan and Elaine Chew, "Fuzzy Analysis in Pitch Class Determination for Polyphonic Audio Key Finding," in *Proceedings of the 6th International Society for Music Information Retrieval Conference (ISMIR)*, 2005, pp. 296–303.
- [39] Ching-Hua Chuan and Elaine Chew, "Polyphonic Audio Key Finding Using the Spiral Array CEG Algorithm," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, 2005, pp. 21–24.
- [40] Henry Leland Clarke, "Toward a Musical Periodization of Music," *Journal of the American Musicological Society*, vol. 9, no. 1, pp. 25–30, 1956.
- [41] Richard L. Cohn, "Neo-Riemannian Operations, Parsimonious Trichords, and their "Tonnetz" Representations," *Journal of Music Theory*, vol. 41, no. 1, pp. 1–66, 1997.
- [42] Richard L. Cohn, *Audacious Euphony*, Oxford University Press, Oxford, 2012.
- [43] James W. Cooley and John W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Mathematics of Computation*, vol. 19, no. 90, pp. 297–301, 1965.

- [44] Corinna Cortes and Vladimir N. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [45] Carl Dahlhaus, Julian Anderson, Charles Wilson, Richard L. Cohn, and Brian Hyer, "Harmony," in *Grove Music Online: Oxford Music Online*, Deane Root, Ed. Oxford University Press, 2001.
- [46] Roger B. Dannenberg, Belinda Thom, and David Watson, "A Machine Learning Approach to Musical Style Recognition," in *Proceedings of the International Computer Music Conference (ICMC)*, 1997.
- [47] W. Bas de Haas, José Pedro Magalhães, Remco C. Veltkamp, and Frans Wiering, "HARMTRACE: Improving Harmonic Similarity Estimation Using Functional Harmony Analysis," in *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, 2011, pp. 67–72.
- [48] W. Bas de Haas, José Pedro Magalhaes, Frans Wiering, and Remco C. Veltkamp, "Automatic Functional Harmonic Analysis," *Computer Music Journal*, vol. 37, no. 4, pp. 37–53, 2014.
- [49] W. Bas de Haas, Martin Rohrmeier, Remco C. Veltkamp, and Frans Wiering, "Modeling Harmonic Similarity Using a Generative Grammar of Tonal Harmony," in *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, 2009, pp. 549–554.
- [50] W. Bas de Haas, Remco C. Veltkamp, and Frans Wiering, "Tonal Pitch Step Distance: A Similarity Measure for Chord Progressions," in *Proceedings of the 9th International Society for Music Information Retrieval Conference (ISMIR)*, 2008, pp. 51–56.
- [51] Diether de la Motte, *Harmonielehre*, Bärenreiter, Kassel, 1976.
- [52] Diether de la Motte and Jeffrey L. Prater, *The Study of Harmony: An Historical Perspective: English translation*, William C. Brown Pub., Dubuque (Iowa), 1991.
- [53] Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin, "Maximum Likelihood From Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, pp. 1–38, 1977.
- [54] Richard Desper and Olivier Gascuel, "Fast and Accurate Phylogeny Reconstruction Algorithms Based on the Minimum-Evolution Principle," *Journal of Computational Biology*, vol. 9, no. 5, pp. 687–705, 2002.
- [55] Simon Dixon, Elias Pampalk, and Gerhard Widmer, "Classification of Dance Music by Periodicity Patterns," in *Proceedings of the 4th International Society for Music Information Retrieval Conference (ISMIR)*, 2003.
- [56] Ofer Dor and Yoram Reich, "An Evaluation of Musical Score Characteristics for Automatic Classification of Composers," *Computer Music Journal*, vol. 35, no. 3, pp. 86–97, 2011.
- [57] Daniel P. W. Ellis and Graham E. Poliner, "Identifying 'Cover Songs' with Chroma Features and Dynamic Programming Beat Tracking," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2007, vol. 4, pp. 1429–1432.
- [58] Daniel P. W. Ellis and Adrian V. Weller, "The 2010 LABROSA Chord Recognition System," in *Music Information Retrieval Evaluation eXchange (MIREX) System Abstracts*. 2010.
- [59] Sebastian Ewert, Meinard Müller, Verena Konz, Daniel Müllensiefen, and Geraint Wiggins, "Towards Cross-Version Harmonic Analysis of Music," *IEEE Transactions on Multimedia*, vol. 14, no. 3, pp. 770–782, 2012.
- [60] Hugo Fastl and Eberhard Zwicker, *Psychoacoustics: Facts and Models*, Springer, Berlin and Heidelberg, 1990.
- [61] Tom Fawcett, "ROC Graphs: Notes and Practical Considerations for Researchers," *Machine Learning*, vol. 31, pp. 1–38, 2004.
- [62] Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, and Dinani Amorim, "Do we Need Hundreds of Classifiers to Solve Real World Classification Problems?," *Journal of Machine Learning Research*, vol. 15, pp. 3133–3181, 2014.
- [63] Arthur Flexer, "A Closer Look on Artist Filters for Musical Genre Classification," in *Proceedings of the 8th International Society for Music Information Retrieval Conference (ISMIR)*, 2007, pp. 341–344.
- [64] Allen Forte, *The Structure of Atonal Music*, Yale University Press, New Haven and London, 1973.
- [65] Paul L. Frank, "Historical or Stylistic Periods?," *Journal of Aesthetics and Art Criticism*, vol. 13, no. 4, pp. 451–457, 1955.
- [66] Wilhelm Fucks and Josef Lauter, *Exaktwissenschaftliche Musikanalyse*, Westdeutscher Verlag, Köln and Opladen, 1965.

- [67] Takuya Fujishima, “Realtime Chord Recognition of Musical Sound: A System Using Common Lisp Music,” in *Proceedings of the International Computer Music Conference (ICMC)*, 1999, pp. 464–467.
- [68] Dennis Gabor, “Theory of Communication: Part 1: The Analysis of Information,” *Journal of the Institution of Electrical Engineers-Part III: Radio and Communication Engineering*, vol. 93, no. 26, pp. 429–441, 1946.
- [69] Zsolt Gárdonyi and Hubert Nordhoff, *Harmonik*, Möseler, Wolfenbüttel, 2nd edition, 2002.
- [70] Daniel Gärtner, Christoph Zipperle, and Christian Dittmar, “Classification of Electronic Club-Music,” in *Proceedings of the DAGA 2010: 36. Jahrestagung für Akustik*, 2010.
- [71] Gabriel Gatzsche and Markus Mehnert, *Ein Beitrag zur tonraumbasierten Analyse und Synthese musikalischer Audiosignale*, PhD thesis, Technische Universität Ilmenau, Ilmenau, 2011.
- [72] Gabriel Gatzsche, Markus Mehnert, David Gatzsche, and Karlheinz Brandenburg, “A Symmetry Based Approach for Musical Tonality Analysis,” in *Proceedings of the 8th International Society for Music Information Retrieval Conference (ISMIR)*, 2007, pp. 207–210.
- [73] Jeroen Geertzen and Menno van Zaanen, “Composer Classification Using Grammatical Inference,” in *Proceedings of the MLM International Workshop on Machine Learning and Music*, 2008, pp. 17–18.
- [74] Irving Godt, “Style Periods of Music History Considered Analytically,” *College Music Symposium*, vol. 24, pp. 33–48, 1984.
- [75] Emilia Gómez, “Key Estimation from Polyphonic Audio,” in *Proceedings of the 1st Annual Music Information Retrieval Evaluation eXchange (MIREX '05)*, 2005.
- [76] Emilia Gómez, *Tonal Description of Music Audio Signals*, PhD thesis, Universitat Pompeu Fabra, Barcelona, 2006.
- [77] Emilia Gómez, “Tonal Description of Polyphonic Audio for Music Content Processing,” *INFORMS Journal on Computing*, vol. 18, no. 3, pp. 294–304, 2006.
- [78] Emilia Gómez and Perfecto Herrera, “Estimating The Tonality Of Polyphonic Audio Files: Cognitive Versus Machine Learning Modelling Strategies,” in *Proceedings of the 5th International Society for Music Information Retrieval Conference (ISMIR)*, 2004.
- [79] Michael Good, “MusicXML for Notation and Analysis,” *Computing in Musicology*, vol. 12, pp. 113–124, 2001.
- [80] Robert Gräfe, *Automatische Analyse und Klassifizierung von Audiodaten anhand von Tonartverläufen*, Bachelor’s Thesis, Technische Universität Ilmenau, Ilmenau, 2015.
- [81] Peter Grosche, Meinard Müller, and Joan Serrà, “Audio Content-Based Music Retrieval,” in *Multimodal Music Processing*, Meinard Müller, Masataka Goto, and Markus Schedl, Eds., vol. 3 of *Dagstuhl Follow-Ups*, pp. 157–174. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2012.
- [82] Bernhard Haas, *Die neue Tonalität von Schubert bis Webern: Hören und Analysieren nach Albert Simon*, Veröffentlichungen zur Musikforschung. F. Noetzel, Wilhelmshaven, 2004.
- [83] Julian Habryka and Christof Weiß, “Zum Scherzo aus Hans Rotts 1. Sinfonie,” in *Mythos Handwerk?*, Ariane Jeßulat, Ed., pp. 187–212. Königshausen & Neumann, Würzburg, 2015.
- [84] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten, “The WEKA Data Mining Software: An Update,” *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.
- [85] Philippe Hamel, “Pooled Features Classification MIREX 2011 Submission,” in *Proceedings of the 7th Annual Music Information Retrieval Evaluation eXchange (MIREX '11)*, 2011.
- [86] Howard Hanson, *Harmonic Materials of Modern Music: Resources of the Tempered Scale*, Appleton-Century-Crofts, New York, 1960.
- [87] Christopher A. Harte and Mark Sandler, “Automatic Chord Identification Using a Quantised Chromagram,” in *Proceedings of the 118th AES Convention*, 2005.
- [88] Paul H. Harvey and Mark D. Pagel, *The Comparative Method in Evolutionary Biology*, vol. 239, Oxford University Press, Oxford, UK, 1991.
- [89] Thomas Hedges, Pierre Roy, and François Pachet, “Predicting the Composer and Style of Jazz Chord Progressions,” *Journal of New Music Research*, vol. 43, no. 3, pp. 276–290, 2014.
- [90] Johann David Heinichen, *Der General-Bass in der Composition*, vol. 2, Dresden, 1728.

- [91] Ruben Hillewaere, Bernard Manderick, and Darrell Conklin, "String Quartet Classification with Monophonic Models," in *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR)*, 2010, pp. 537–542.
- [92] Aline Honingh and Rens Bod, "Pitch Class Set Categories as Analysis Tools for Degrees of Tonality," in *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR)*, 2010, pp. 459–464.
- [93] Aline Honingh and Rens Bod, "Clustering and Classification of Music by Interval Categories," in *Proceedings of the Third International Conference on Mathematics and Computation in Music*, Berlin and Heidelberg, 2011, MCM'11, pp. 346–349, Springer-Verlag.
- [94] Aline Honingh, Tillman Weyde, and Darrell Conklin, "Sequential Association Rules in Atonal Music," in *Mathematics and Computation in Music (MCM)*. 2009, pp. 130–138, Springer.
- [95] Maria Hontanilla, Carlos Pérez-Sancho, and José Manuel Iñesta, "Modeling Musical Style with Language Models for Composer Recognition," in *Pattern Recognition and Image Analysis*, pp. 740–748. Springer, 2013.
- [96] Patrick O. Hoyer, "Non-Negative Matrix Factorization With Sparseness Constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.
- [97] Chi-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, "A Practical Guide to Support Vector Classification," 2003.
- [98] Zhen Hu, Kun Fu, and Changshui Zhang, "Audio Classical Composer Identification by Deep Neural Network," *Computing Research Repository*, 2013.
- [99] David Miles Huber, *The MIDI Manual: A Practical Guide to MIDI in the Project Studio*, Focal Press, Waltham, Massachusetts, 3rd edition, 2007.
- [100] Brian Hyer, "Tonality," in *Grove Music Online: Oxford Music Online*, Deane Root, Ed. Oxford University Press, 2001.
- [101] Plácido R. Illescas, David Rizo, and José Manuel Iñesta, "Harmonic, Melodic, and Functional Automatic Analysis," in *Proceedings of the International Computer Music Conference (ICMC)*, 2007, pp. 165–168.
- [102] Özgür Izmirlı, "An Algorithm For Audio Key Finding," in *Proceedings of the 1st Annual Music Information Retrieval Evaluation eXchange (MIREX '05)*, 2005.
- [103] Özgür Izmirlı, "Template Based Key Finding From Audio," in *Proceedings of the International Computer Music Conference (ICMC)*, 2005, pp. 211–214.
- [104] Özgür Izmirlı, "Audio Key Finding Using Low-Dimensional Spaces," in *Proceedings of the 7th International Society for Music Information Retrieval Conference (ISMIR)*, 2006, pp. 127–132.
- [105] Özgür Izmirlı, "Localized Key Finding From Audio Using Nonnegative Matrix Factorization for Segmentation," in *Proceedings of the 8th International Society for Music Information Retrieval Conference (ISMIR)*, 2007, pp. 195–200.
- [106] Özgür Izmirlı, "Tonal-Atonal Classification of Music Audio Using Diffusion Maps," in *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, Kobe, Japan, 2009, pp. 687–691.
- [107] Anil K. Jain and Balakrishnan Chandrasekaran, "Dimensionality and Sample Size Considerations in Pattern Recognition Practice," in *Handbook of Statistics*, Paruchuri R. Krishnaiah and Laveen Kanal, Eds., vol. 2, pp. 835–855. Elsevier, Amsterdam, 1982.
- [108] Dan-Ning Jiang, Lie Lu, Hong-Jiang Zhang, Jian-Hua Tao, and Lian-Hong Cai, "Music Type Classification by Spectral Contrast Feature," in *Proceedings of the International Conference on Multimedia and Expo (ICME)*, 2002, vol. 1, pp. 113–116.
- [109] Nanzhu Jiang, Peter Grosche, Verena Konz, and Meinard Müller, "Analyzing Chroma Feature Types for Automated Chord Recognition," in *Proceedings of the 42nd AES International Conference on Semantic Audio*, 2011, pp. 285–294.
- [110] Nanzhu Jiang and Meinard Müller, "Automated Methods for Analyzing Music Recordings in Sonata Form," in *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR)*, 2013, pp. 595–600.
- [111] Alexander Kachkaev, Daniel Wolff, Mathieu Barthet, Mark Plumbley, Jason Dykes, and Tillman Weyde, "Visualising Chord Progressions in Music Collections: A Big Data Approach," in *Proceedings of the 9th Conference on Interdisciplinary Musicology (CIM)*, 2014, pp. 180–183.

- [112] Hitomi Kaneko, Daisuke Kawakami, and Shigeki Sagayama, "Functional Harmony Annotation Database for Statistical Music Analysis," in *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR): Late breaking session*, 2010.
- [113] Gary S. Karpinski, "Ambiguity: Another Listen," *Music Theory Online*, vol. 18, no. 3, 2012.
- [114] Edward J. Kessler, Christa Hansen, and Roger N. Shepard, "Tonal Schemata in the Perception of Music in Bali and in the West," *Music Perception: An Interdisciplinary Journal*, vol. 2, no. 2, pp. 131–165, 1984.
- [115] Maksim Khadkevich and Maurizio Omologo, "Reassigned Spectrum-Based Feature Extraction for GMM-Based Automatic Chord Recognition," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2013, no. 1, pp. 1–12, 2013.
- [116] Francis J. Kiernan, "Score-based Style Recognition Using Artificial Neural Networks," in *Proceedings of the 1st International Symposium on Music Information Retrieval (ISMIR)*, 2000.
- [117] Anssi Klapuri, "Multiple Fundamental Frequency Estimation by Summing Harmonic Amplitudes," in *Proceedings of the 7th International Society for Music Information Retrieval Conference (ISMIR)*, 2006, pp. 216–221.
- [118] Anssi Klapuri, "Multipitch Analysis of Polyphonic Music and Speech Signals Using an Auditory Model," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 255–266, 2008.
- [119] Anssi Klapuri, Antti J. Eronen, and Jaakko T. Astola, "Analysis of the Meter of Acoustic Musical Signals," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 342–355, 2006.
- [120] Teuvo Kohonen, "Self-Organized Formation of Topologically Correct Feature Maps," *Biological Cybernetics*, vol. 43, no. 1, pp. 59–69, 1982.
- [121] Verena Konz, Meinard Müller, and Rainer Kleinertz, "A Cross-Version Chord Labelling Approach for Exploring Harmonic Structures—A Case Study on Beethoven's Appassionata," *Journal of New Music Research*, vol. 42, no. 1, pp. 61–77, 2013.
- [122] Stefan M. Kostka, Dorothy Payne, and Byron Almén, *Tonal Harmony*, McGraw-Hill, New York, 7th edition, 2012.
- [123] Ted Kronvall, Maria Juhlin, Stefan I. Adalbjörnsson, and Andreas Jakobsson, "Sparse Chroma Estimation for Harmonic Audio," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015, pp. 579–583.
- [124] Carol L. Krumhansl, *Cognitive Foundations of Musical Pitch*, Oxford Psychology Series. Oxford University Press, 1990.
- [125] Joseph B. Kruskal and Myron Wish, *Multidimensional Scaling*, vol. 11, Sage Publications, Beverly Hills, California, 1978.
- [126] Anna M. Kruspe, Hanna Lukashevich, Jakob Abeßer, Holger Großmann, and Christian Dittmar, "Automatic Classification of Musical Pieces into Global Cultural Areas," in *Proceedings of the 42nd AES International Conference on Semantic Audio*, 2011.
- [127] Steven G. Laitz, *The Complete Musician: An Integrated Approach to Tonal Theory, Analysis, and Listening*, Oxford University Press, New York, 3rd edition, 2011.
- [128] Olivier Lartillot and Petri Toiviainen, "A Toolbox for Musical Feature Extraction From Audio," in *Proceedings of the 8th International Society for Music Information Retrieval Conference (ISMIR)*, 2007.
- [129] Jan LaRue, "On Style Analysis," *Journal of Music Theory*, vol. 6, no. 1, pp. 91–107, 1962.
- [130] Jan LaRue, *Guidelines for Style Analysis*, Harmonie Park Press, Michigan, 1992.
- [131] Kyogu Lee, "Automatic Chord Recognition from Audio Using Enhanced Pitch Class Profile," in *Proceedings of the International Computer Music Conference (ICMC)*, 2006.
- [132] Kyogu Lee and Malcolm Slaney, "Automatic Chord Recognition From Audio Using a Supervised HMM Trained With Audio-From-Symbolic Data," in *Proceedings of the 1st ACM Workshop on Audio and Music Computing Multimedia*, 2006, pp. 11–20.
- [133] Ernő Lendvai, *Symmetrien in der Musik: Einführung in die musikalische Semantik*, Universal Edition, Wien, 1995.
- [134] Ernő Lendvai, Miklós Szabó, and Miklós Mohay, *Symmetries of Music: An Introduction to Semantics of Music*, Kodály Institute, Kecskemét, Hungary, 1993.

- [135] Fred Lerdahl, *Tonal Pitch Space*, Oxford University Press, New York, 2001.
- [136] Fred Lerdahl and Carol L. Krumhansl, "Modeling Tonal Tension," *Music Perception: An Interdisciplinary Journal*, vol. 24, no. 4, pp. 329–366, 2007.
- [137] Joel Lester, "Rameau and Eighteenth-Century Harmonic Theory," in *The Cambridge History of Western Music Theory*, pp. 753–777. Cambridge University Press, Cambridge, 2002.
- [138] Thomas Lidy, Andreas Rauber, A. Pertusa, and José Manuel Iñesta, "Improving Genre Classification by Combination of Audio and Symbolic Descriptors Using a Transcription System," in *Proceedings of the 8th International Society for Music Information Retrieval Conference (ISMIR)*, 2007, pp. 61–66.
- [139] Adam T. Lindsay and Jürgen Herre, "MPEG-7 and MPEG-7 Audio - An Overview," *Journal of Audio Engineering Society*, vol. 49, no. 7/8, pp. 589–594, 2001.
- [140] Stuart P. Lloyd, "Least Squares Quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.
- [141] Beth Logan, "Mel Frequency Cepstral Coefficients for Music Modeling," in *Proceedings of the 1st International Symposium on Music Information Retrieval (ISMIR)*, 2000.
- [142] Beth Logan and Ariel Salomon, "A Music Similarity Function Based on Signal Analysis," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, Tokyo and Japan, 2001, IEEE Computer Society.
- [143] Alfred Lorenz, *Das Geheimnis der Form bei Richard Wagner: Der musikalische Aufbau des Bühnenfestspiels "Der Ring des Nibelungen"*, Berlin, 1924.
- [144] James MacQueen, "Some Methods for Classification and Analysis of Multivariate Observations," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1967, vol. 1, pp. 281–297.
- [145] Wilhelm Maler, *Beitrag zur durmolltonalen Harmonielehre*, Leuckart, München and Leipzig, 13th edition, 1984.
- [146] Aleix M. Martínez and Avinash C. Kak, "PCA versus LDA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228–233, 2001.
- [147] Matthias Mauch and Simon Dixon, "Approximate Note Transcription for the Improved Identification of Difficult Chords," in *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR)*, 2010, pp. 135–140.
- [148] Matthias Mauch and Simon Dixon, "Simultaneous Estimation of Chords and Musical Context From Audio," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 6, pp. 1280–1289, 2010.
- [149] Matthias Mauch, Simon Dixon, Christopher Harte, Michael Casey, and Benjamin Fields, "Discovering Chord Idioms Through Beatles and Real Book Songs," in *Proceedings of the 8th International Society for Music Information Retrieval Conference (ISMIR)*, 2007, pp. 255–258.
- [150] Matthias Mauch and Mark Levy, "Structural Change on Multiple Time Scales as a Correlate of Musical Complexity," in *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, 2011, pp. 489–494.
- [151] Matthias Mauch, Robert M. MacCallum, Mark Levy, and Armand M. Leroi, "The Evolution of Popular Music: USA 1960–2010," *Royal Society Open Science*, vol. 2, no. 5, 2015.
- [152] Warren S. McCulloch and Walter Pitts, "A Logical Calculus of the Ideas Immanent in Nervous Activity," *The Bulletin of Mathematical Biophysics*, vol. 5, no. 4, pp. 115–133, 1943.
- [153] Cory McKay and Ichiro Fujinaga, "Automatic Genre Classification Using Large High-Level Musical Feature Sets," in *Proceedings of the 5th International Society for Music Information Retrieval Conference (ISMIR)*, 2004, pp. 525–530.
- [154] Martin F. McKinney and Jeroen Breebaart, "Features for Audio and Music Classification," in *Proceedings of the 4th International Society for Music Information Retrieval Conference (ISMIR)*, 2003.
- [155] Lesley Mearns, Emmanouil Benetos, and Simon Dixon, "Automatically Detecting Key Modulations in J. S. Bach Chorale Recordings," in *Proceedings of the 8th Sound and Music Computing Conference (SMC)*, 2011, pp. 25–32.
- [156] Lesley Mearns, Dan Tidhar, and Simon Dixon, "Characterisation of Composer Style Using High-level Musical Features," in *Proceedings of the 3rd International Workshop on Machine Learning and Music (MML)*, 2010, pp. 37–40.

- [157] Markus Mehnert, Gabriel Gatzsche, and Daniel Arndt, “Symmetry Model Based Key Finding,” in *Proceedings of the 126th AES Convention*, 2009.
- [158] Paul Mermelstein, “Distance measures for speech recognition, psychological and instrumental,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 116, pp. 374–388, 1976.
- [159] Olivier Messiaen, *The Technique of My Musical Language*, Leduc, Paris, 1944.
- [160] Jean Molino, J. A. Underwood, and Craig Ayrey, “Musical Fact and the Semiology of Music,” *Music Analysis*, pp. 105–156, 1990.
- [161] Meinard Müller, *Information Retrieval for Music and Motion*, Springer, Berlin and Heidelberg, 2007.
- [162] Meinard Müller, *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*, Springer, Berlin and Heidelberg, 2015.
- [163] Meinard Müller, Daniel P. W. Ellis, Anssi Klapuri, and Gaël Richard, “Signal Processing for Music Analysis,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 6, pp. 1088–1110, 2011.
- [164] Meinard Müller and Sebastian Ewert, “Towards Timbre-Invariant Audio Features for Harmony-Based Music,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 649–662, 2010.
- [165] Meinard Müller and Sebastian Ewert, “Chroma Toolbox: MATLAB Implementations for Extracting Variants of Chroma-Based Audio Features,” in *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, 2011, pp. 215–220.
- [166] Meinard Müller, Sebastian Ewert, and Sebastian Kreuzer, “Making Chroma Features More Robust to Timbre Changes,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 1877–1880.
- [167] Meinard Müller, Frank Kurth, and Michael Clausen, “Chroma-Based Statistical Audio Features for Audio Matching,” in *Proceedings Workshop on Applications of Signal Processing (WASPAA)*, 2005, pp. 275–278.
- [168] Meinard Müller and Nanzhu Jiang, “A Scape Plot Representation for Visualizing Repetitive Structures of Music Recordings,” in *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR)*, 2012, pp. 97–102.
- [169] Meinard Müller, Verena Konz, Wolfgang Bogler, and Vlora Arifi-Müller, “Saarland Music Data,” in *Late-Breaking and Demo Session of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, 2011.
- [170] Meinard Müller, Frank Kurth, and Michael Clausen, “Audio Matching via Chroma-Based Statistical Features,” in *Proceedings of the 6th International Society for Music Information Retrieval Conference (ISMIR)*, 2005, pp. 288–295.
- [171] Jean-Jacques Nattiez, *Music and Discourse: Toward a Semiology of Music*, Princeton University Press, 1990.
- [172] Andrew Y. Ng and Michael I. Jordan, “On Discriminative vs. Generative Classifiers: A Comparison of Logistic Regression and Naive Bayes,” in *Advances in Neural Information Processing Systems 14*, Thomas G. Dietterich, Suzanna Becker, and Zoubin Ghahramani, Eds. 2001, pp. 841–848, MIT Press.
- [173] Yizhao Ni, Matt McVicar, Raúl Santos-Rodríguez, and Tjil de Bie, “An End-to-End Machine Learning System for Harmonic Analysis of Music,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1771–1783, 2012.
- [174] Katy Noland and Mark Sandler, “Influences of Signal Processing, Tone Profiles, and Chord Progressions on a Model for Estimating the Musical Key From Audio,” *Computer Music Journal*, vol. 33, no. 1, pp. 42–56, 2009.
- [175] Mitsunori Ogihara and Tao Li, “N-Gram Chord Profiles for Composer Style Identification,” in *Proceedings of the 9th International Society for Music Information Retrieval Conference (ISMIR)*, 2008, pp. 671–676.
- [176] Nobutaka Ono, Kenichi Miyamoto, Jonathan Kameoka Hirokazu Le Roux, and Shigeki Sagayama, “Separation of a Monaural Audio Signal Into Harmonic/Percussive Components by Complementary Diffusion on Spectrogram,” in *Proceedings of the 16th European Signal Processing Conference (EUSIPCO)*, 2008, pp. 1–4.

- [177] Jean-François Paiement, Douglas Eck, and Samy Bengio, “A Probabilistic Model for Chord Progressions,” in *Proceedings of the 6th International Society for Music Information Retrieval Conference (ISMIR)*, 2005, pp. 312–319.
- [178] Elias Pampalk, Arthur Flexer, and Gerhard Widmer, “Improvements of Audio-Based Music Similarity and Genre Classification,” in *Proceedings of the 6th International Society for Music Information Retrieval Conference (ISMIR)*, 2005, pp. 628–633.
- [179] Maria Panteli, Emmanouil Benetos, and Simon Dixon, “Learning a feature space for similarity in world music,” in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, New York, USA, 2016, pp. 538–544.
- [180] Hélène Papadopoulos and Geoffroy Peeters, “Local Key Estimation From an Audio Signal Relying on Harmonic and Metrical Structures,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 4, pp. 1297–1312, 2012.
- [181] Mitchell Parry, “Musical Complexity and Top 40 Chart Performance: Technical Report,” 2004.
- [182] Robert Pascall, “Style,” in *Grove Music Online: Oxford Music Online*, Deane Root, Ed. Oxford University Press, 2001.
- [183] Steffen Pauws, “Musical Key Extraction From Audio,” in *Proceedings of the 5th International Society for Music Information Retrieval Conference (ISMIR)*, 2004.
- [184] Geoffroy Peeters, “A Large Set of Audio Features for Sound Description (Similarity and Classification) in the CUIDADO Project: Technical Report,” 2004.
- [185] Geoffroy Peeters, “Chroma-Based Estimation of Musical Key From Audio-Signal Analysis,” in *Proceedings of the 7th International Society for Music Information Retrieval Conference (ISMIR)*, 2006, pp. 115–120.
- [186] Geoffroy Peeters, “Musical Key Estimation of Audio Signals Based on Hidden Markov Modeling of Chroma Vectors,” in *Proceedings of the 9th International Conference on Digital Audio Effects (DAFx)*, 2006.
- [187] Geoffroy Peeters and Xavier Rodet, “Hierarchical Gaussian Tree with Inertia Ratio Maximization for the Classification of Large Musical Instruments Databases,” in *Proceedings of the 6th International Conference on Digital Audio Effects (DAFx)*, 2003.
- [188] C. Pérez-Sancho, D. Rizo, and José Manuel Iñesta, “Genre Classification Using Chords and Stochastic Language Models,” *Connection Science*, vol. 21, no. 2 & 3, pp. 145–159, 2009.
- [189] Carlos Pérez-Sancho, D. Rizo, José Manuel Iñesta, Pedro José Ponce de León, S. Kersten, and Rafael Ramirez, “Genre Classification of Music by Tonal Harmony,” *Intelligent Data Analysis*, vol. 14, no. 5, pp. 533–545, 2010.
- [190] Daniel Perttu, “A Quantitative Study of Chromaticism: Changes Observed in Historical Eras and Individual Composers,” *Empirical Musicology Review*, vol. 2, no. 2, pp. 47–54, 2007.
- [191] Walter Piston, *Harmony*, Norton, New York, 1941.
- [192] Pedro José Ponce de León and José Manuel Iñesta, “Musical Style Classification from Symbolic Data: A Two Styles Case Study,” *Selected Papers from the Proceedings of the Computer Music Modeling and Retrieval 2003, Lecture Notes in Computer Science*, vol. 2771, pp. 167–177, 2004.
- [193] Pedro José Ponce de León and José Manuel Iñesta, “A Pattern Recognition Approach For Music Style Identification Using Shallow Statistical Descriptors,” *IEEE Transactions on System, Man and Cybernetics - Part C : Applications and Reviews*, vol. 37, no. 2, pp. 248–257, 2007.
- [194] Hendrik Purwins, Benjamin Blankertz, and Klaus Obermayer, “Constant Q Profiles for Tracking Modulations in Audio Data,” in *Proceedings of the 2001 International Computer Music Conference (ICMC)*, 2001.
- [195] Hendrik Purwins, Benjamin Blankertz, Klaus Obermayer, and Guido Dornhege, “Scale Degree Profiles From Audio Investigated With Machine Learning,” in *Proceedings of the 116th Audio Engineering Society (AES) Convention*, 2004.
- [196] Ian Quinn, “Listening to Similarity Relations,” *Perspectives of New Music*, vol. 39, no. 2, pp. 108–158, 2001.
- [197] Jean Philippe Rameau, *Generation Harmonique ou Traité de Musique Theorique et Pratique*, Prault fils, Paris, 1737.

- [198] Sarunas J. Raudys and Anil K. Jain, “Small Sample Size Effects in Statistical Pattern Recognition: Recommendations for Practitioners,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 3, pp. 252–264, 1991.
- [199] Rudolph Reti, *Tonality, Atonality, Pantonality: A Study of Some Trends in Twentieth Century Music*, Rockliff, London, 1958.
- [200] Hugo Riemann, *Vereinfachte Harmonielehre oder die Lehre von den tonalen Funktionen der Akkorde*, Augener, London, 1893.
- [201] Thomas Rocher, Matthias Robine, Pierre Hanna, and Laurent Oudre, “Concurrent Estimation of Chords and Keys From Audio,” in *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR)*, 2010, pp. 141–146.
- [202] Pablo H. Rodriguez Zivic, Favio Shifres, and Guillermo A. Cecchi, “Perceptual Basis of Evolving Western Musical Styles,” *Proceedings of the National Academy of Sciences*, vol. 110, no. 24, pp. 10034–10038, 2013.
- [203] Martin Rohrmeier, “Towards a Generative Syntax of Tonal Harmony,” *Journal of Mathematics and Music*, vol. 5, no. 1, pp. 35–53, 2011.
- [204] Miguel A. Roig-Francolí, *Harmony in Context*, McGraw-Hill Humanities/Social Sciences/Languages, New York, 2nd edition, 2011.
- [205] Charles Rosen, *The Classical Style: Haydn, Mozart, Beethoven*, W. W. Norton, 1971.
- [206] Herbert Rosenberg, “On the Analysis of Style,” *Acta Musicologica*, vol. 9, pp. 5–11, 1937.
- [207] Peter J. Rousseeuw, “Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis,” *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.
- [208] Matti P. Rynänen and Anssi Klapuri, “Automatic Transcription of Melody, Bass Line, and Chords in Polyphonic Music,” *Computer Music Journal*, vol. 32, pp. 72–86, 2008.
- [209] Craig Stuart Sapp, “Harmonic Visualizations of Tonal Music,” in *Proceedings of the 2001 International Computer Music Conference (ICMC)*, 2001.
- [210] Craig Stuart Sapp, “Visual Hierarchical Key Analysis,” *ACM Computers in Entertainment*, vol. 3, no. 4, pp. 1–19, 2005.
- [211] Maximilian Schaab, *Automatische Klassifikation klassischer Musikstile anhand relativer Tonhöhenklassen*, Bachelor’s Thesis, Technische Universität Ilmenau, Ilmenau, 2015.
- [212] Heinrich Schenker, *Neue musikalische Theorien und Phantasien I: Harmonielehre*, Cotta, Stuttgart and Berlin, 1906.
- [213] Ricardo Scholz, Emmanuel Vincent, and Frédéric Bimbot, “Robust Modeling of Musical Chord Sequences Using Probabilistic N-Grams,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 53–56.
- [214] Arnold Schönberg, *Harmonielehre*, Universal Edition, Wien, 1922.
- [215] Arnold Schönberg, *Theory of Harmony: Translated by Roy E. Carter*, University of California Press, Berkeley, 1983.
- [216] Arnold Schönberg and Leonard Stein, *Style and Idea: Selected Writings of Arnold Schoenberg*, St. Martins Press, New York, 1975.
- [217] Christian Schörkhuber and Anssi Klapuri, “Constant-Q Transform Toolbox for Music Processing,” in *Proceedings of the 7th Sound and Music Computing Conference (SMC)*, 2010, pp. 3–64.
- [218] Björn Schuller and Benedikt Gollan, “Music Theoretic and Perception-based Features for Audio Key Determination,” *Journal of New Music Research*, vol. 41, no. 2, pp. 175–193, 2012.
- [219] Simon Sechter, *Die Grundsätze der musikalischen Komposition*, vol. 3, Breitkopf & Härtel, Leipzig, 1853.
- [220] Desmond Sergeant, “The Octave—Percept or Concept,” *Psychology of Music*, vol. 11, no. 1, pp. 3–18, 1983.
- [221] Alexander Sheh and Daniel P. W. Ellis, “Chord Segmentation and Recognition using EM-Trained Hidden Markov Models,” in *Proceedings of the 4th International Society for Music Information Retrieval Conference (ISMIR)*, 2003.

- [222] Arun Shenoy, Roshni Mohapatra, and Ye Wang, “Key Determination of Acoustic Musical Signals,” in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, Edinburgh, Scotland, UK, 2004, vol. 3, pp. 1771–1774.
- [223] Roger N. Shepard, “Circularity in Judgments of Relative Pitch,” *Journal of the Acoustical Society of America*, vol. 36, no. 12, pp. 2346–2353, 1964.
- [224] Roger N. Shepard, “Geometrical Approximations to the Structure of Musical Pitch,” *Psychological Review*, vol. 89, no. 4, pp. 305, 1982.
- [225] Christian Simmermacher, Da Deng, and Stephen Cranefield, “Feature Analysis and Classification of Classical Musical Instruments: An Empirical Study,” in *Advances in Data Mining. Applications in Medicine, Web Mining, Marketing, Image and Signal Mining*, vol. 4065 of *Lecture Notes in Computer Science*, pp. 444–458. Springer, Berlin and Heidelberg, 2006.
- [226] Bryan Simms, “Choron, Fétis, and the Theory of Tonality,” *Journal of Music Theory*, vol. 19, no. 1, pp. 112–138, 1975.
- [227] Nicholas Slonimsky, “Notation,” in *Baker’s Dictionary of Music*, Richard Kassel, Ed., pp. 718–721. Schirmer Reference, New York, 1997.
- [228] Michael Stein, B. M. Schubert, Matthias Gruhne, Gabriel Gatzsche, and Markus Mehnert, “Evaluation and Comparison of Audio Chroma Feature Extraction Methods,” in *Proceedings of the 126th AES Convention*, 2009.
- [229] Stanley S. Stevens, John Volkman, and Edwin B. Newman, “A Scale for the Measurement of the Psychological Magnitude Pitch,” *Journal of the Acoustical Society of America*, vol. 8, no. 3, pp. 185–190, 1937.
- [230] Sebastian Streich, *Music Complexity: A Multi-Faceted Description of Audio Content*, PhD Thesis, Universitat Pompeu Fabra, Barcelona, 2006.
- [231] Sebastian Streich and Perfecto Herrera, “Towards Describing Perceived Complexity of Songs: Computational Methods and Implementation,” in *Proceedings of the 25th International AES Conference on Metadata for Audio*, 2004.
- [232] Bob L. Sturm, “Two Systems for Automatic Music Genre Recognition: What Are They Really Recognizing?,” in *Proceedings of the Second International ACM Workshop on Music Information Retrieval with User-Centered and Multimodal Strategies*, 2012, pp. 69–74.
- [233] Bob L. Sturm, “Classification Accuracy Is Not Enough,” *Journal of Intelligent Information Systems*, vol. 41, no. 3, pp. 371–406, 2013.
- [234] Bob L. Sturm and Pardis Noorzad, “On Automatic Music Genre Recognition by Sparse Representation Classification Using Auditory Temporal Modulations,” in *Proceedings of the 9th International Symposium on Computer Music Modeling and Retrieval (CMMR)*, 2012, pp. 379–394.
- [235] David Temperley, *The Cognition of Basic Musical Structures*, MIT Press, 2001.
- [236] Valeri Tsatsishvili, *Automatic Subgenre Classification of Heavy Metal Music*, Master’s thesis, University of Jyväskylä, Jyväskylä, 2011.
- [237] George Tzanetakis and Perry Cook, “Musical Genre Classification of Audio Signals,” *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.
- [238] Yushi Ueda, Yuki Uchiyama, Takuya Nishimoto, Nobutaka Ono, and Shigeki Sagayama, “HMM-Based Approach for Automatic Chord Detection Using Refined Acoustic Features,” in *Proceedings of the IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, 2010, pp. 5518–5521.
- [239] Steven van de Par, Martin F. McKinney, and André Redert, “Musical Key Extraction From Audio Using Profile Training,” in *Proceedings of the 7th International Society for Music Information Retrieval Conference (ISMIR)*, 2006, pp. 328–329.
- [240] Donald H. van Ess, *The Heritage of Musical Style: Revised Edition*, University Press of America, Lanham, Maryland, 2007.
- [241] Peter van Kranenburg, “Composer Attribution by Quantifying Compositional Strategies,” in *Proceedings of the 7th International Society for Music Information Retrieval Conference (ISMIR)*, 2006, pp. 375–376.
- [242] Peter van Kranenburg and Eric Backer, “Musical Style Recognition - a Quantitative Approach,” in *Proceedings of the Conference on Interdisciplinary Musicology (CIM)*, 2004, pp. 106–107.
- [243] Vladimir N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, 1995.

- [244] Edgard Varèse and Chou Wen-Chung, “The Liberation of Sound,” *Perspectives of New Music*, pp. 11–19, 1966.
- [245] Michele Ventura, “Detection of Historical Period in Symbolic Music Text,” *International Journal of e-Education, e-Business, e-Management and e-Learning*, vol. 4, no. 1, pp. 32–36, 2014.
- [246] Michel Verleysen and Damien François, “The Curse of Dimensionality in Data Mining and Time Series Prediction,” in *Computational Intelligence and Bioinspired Systems*, pp. 758–770. Springer, Berlin, Heidelberg, 2005.
- [247] Vladimir Viro, “Peachnote: Music Score Search and Analysis Platform,” in *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, 2011, pp. 359–362.
- [248] Andrew R. Webb, *Statistical Pattern Recognition*, John Wiley & Sons, 2nd edition, 2002.
- [249] Jacob Gottfried Weber, *Versuch einer geordneten Theorie der Tonsetzkunst*, vol. 3, B. Schott’s Söhne, Mainz, 1832.
- [250] James Webster, “The Eighteenth Century as a Music-Historical Period?,” *Eighteenth Century Music*, vol. 1, no. 01, pp. 47–60, 2004.
- [251] Claus Weihs, Uwe Ligges, Fabian Mörchen, and Daniel Müllensiefen, “Classification in Music Research,” *Advances in Data Analysis and Classification*, vol. 1, no. 3, pp. 255–291, 2007.
- [252] Christof Weiß, “Global Key Extraction from Classical Music Audio Recordings Based on the Final Chord,” in *Proceedings of the 10th Sound and Music Computing Conference (SMC)*, 2013, pp. 742–747.
- [253] Christof Weiß, Estefanía Cano, and Hanna Lukashevich, “A Mid-Level Approach to Local Tonality Analysis: Extracting Key Signatures from Audio,” in *Proceedings of the 53rd AES International Conference on Semantic Audio*, 2014.
- [254] Christof Weiß and Julian Habryka, “Chroma-Based Scale Matching for Audio Tonality Analysis,” in *Proceedings of the 9th Conference on Interdisciplinary Musicology (CIM)*, 2014, pp. 168–173.
- [255] Christof Weiß, Rainer Kleinertz, and Meinard Müller, “Möglichkeiten der computergestützten Erkennung und Visualisierung harmonischer Strukturen – eine Fallstudie zu Richard Wagners ‘Die Walküre’,” in *Bericht zur Jahrestagung der Gesellschaft für Musikforschung (GfM) 2015 in Halle/Saale*, Wolfgang Auhagen and Wolfgang Hirschmann, Eds., Mainz, Germany, 2016, Schott Campus.
- [256] Christof Weiß, Matthias Mauch, and Simon Dixon, “Timbre-Invariant Audio Features for Style Analysis of Classical Music,” in *Proceedings of the Joint Conference 40th ICMC and 11th SMC*, 2014, pp. 1461–1468.
- [257] Christof Weiß and Meinard Müller, “Quantifying and Visualizing Tonal Complexity,” in *Proceedings of the 9th Conference on Interdisciplinary Musicology (CIM)*, 2014, pp. 184–187.
- [258] Christof Weiß and Meinard Müller, “Tonal Complexity Features for Style Classification of Classical Music,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015, pp. 688–692.
- [259] Christof Weiß and Maximilian Schaab, “On the Impact of Key Detection Performance for Identifying Classical Music Styles,” in *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, 2015, pp. 45–51.
- [260] Adrian Weller, Daniel P. W. Ellis, and Tony Jebara, “Structured Prediction Models for Chord Transcription of Music Audio,” in *Machine Learning and Applications, 2009. ICMLA '09. International Conference on*, 2009, pp. 590–595.
- [261] Wikimedia Commons, “Circle of Fifths: http://commons.wikimedia.org/wiki/File:Circle_of_fifths..._deluxe_4.svg,” 21.03.2015.
- [262] John Wright, Allen Y. Yang, Arvind Ganesh, Shankar S. Sastry, and Yi Ma, “Robust Face Recognition via Sparse Representation,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.
- [263] Iannis Xenakis, *Formalized Music: Thought and Mathematics in Composition*, Pendragon Press, Hillsdale, New York, 1992.
- [264] Changsheng Xu, Namunu C. Maddage, and Xi Shao, “Automatic music classification and summarization,” *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 3, pp. 441–450, 2005.
- [265] Kazuyoshi Yoshii and Masataka Goto, “A Vocabulary-Free Infinity-Gram Model for Nonparametric Bayesian Chord Progression Analysis,” in *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, 2011, pp. 645–650.

-
- [266] Yongwei Zhu and Mohan S. Kankanhalli, “Music Scale Modeling for Melody Matching,” in *Proceedings of the 11th ACM international conference on Multimedia*, New York, 2003, pp. 359–362.
- [267] Yongwei Zhu and Mohan S. Kankanhalli, “Key-Based Melody Segmentation for Popular Songs,” in *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, Cambridge and UK, 2004, vol. 3, pp. 862–865.
- [268] Yongwei Zhu and Mohan S. Kankanhalli, “Precise Pitch Profile Feature Extraction From Musical Audio for Key Detection,” *IEEE Transactions on Multimedia*, vol. 8, no. 3, pp. 575–584, 2006.
- [269] Yongwei Zhu, Mohan S. Kankanhalli, and Sheng Gao, “Music Key Detection for Musical Audio,” in *Proceedings of the 11th IEEE International Multimedia Modelling Conference*, 2005, pp. 30–37.
- [270] Udo Zölzer, *Digital Audio Signal Processing*, John Wiley & Sons, Hoboken, New Jersey, 2nd edition, 2008.
- [271] Eberhard Zwicker, “Subdivision of the Audible Frequency Range Into Critical Bands (Frequenzgruppen),” *Journal of the Acoustical Society of America*, , no. 33 (2), pp. 248, 1961.

List of Figures

2.1	Harmonic series including the first 16 partials of C2.	11
2.2	Shepard's helix of pitch perception.	12
2.3	Pitch classes as a series of perfect fifths.	13
2.4	Generic intervals for the C major scale in relation to C4.	13
2.5	Specific names of intervals and their complementaries.	14
2.6	Chromatic scale in a perfect fifth ordering.	17
2.7	Diatonic modes.	18
2.8	C major scale with scale degree numbers.	19
2.9	Different versions of the C minor scale.	20
2.10	Several non-diatonic scales based on C.	20
2.11	Basic triad types above C4.	21
2.12	Triad inversions shown for the CM triad.	22
2.13	Five seventh chord types used in Western classical music.	23
2.14	Opening choral from J. S. Bach's motet "Jesu, meine Freude."	24
2.15	Scalar triads of the major and minor scales.	26
2.16	Circle of fifths for musical keys.	29
3.1	Overture from L. van Beethoven's opera "Fidelio" op. 72c.	36
3.2	Piano reduction of the "Fidelio" score page.	37
3.3	MusicXML encoding of the Violin I part from Beethoven's "Fidelio" overture.	38
3.4	Piano roll representation of a MIDI file	39
3.5	Waveforms of two audio recordings of Beethoven's "Fidelio" overture (Measures 1–8).	40
3.6	Hamming window function.	42
3.7	Magnitude spectrograms of the two "Fidelio" audio recordings.	43
3.8	Frequency mapping using different scales.	45
3.9	Audio spectral envelope features for the "Fidelio" examples.	46
3.10	Mel scale mapping and triangular filters.	47
3.11	Schematic overview of the MFCC calculation.	47
3.12	Loudness features for the "Fidelio" orchestra excerpt.	47
3.13	Log-frequency spectrograms of the two "Fidelio" examples.	49
3.14	Chromagrams of the two "Fidelio" recordings.	52
3.15	Different chromagram representations of the "Fidelio" orchestra recording, first measures.	56
3.16	Chromagram in different temporal resolutions for the "Fidelio" orchestra recording, first measures.	58
3.17	Chroma histograms of the two "Fidelio" recordings.	59
3.18	Three-fold cross validation.	62
3.19	Gaussian Mixture Model.	63
4.1	Overview of tonality and style analysis tasks.	68
4.2	Hierarchical nature of tonal structures.	69
4.3	Different levels of music genre classification.	73
5.1	Overview of the key extraction process.	78
5.2	A diatonic subset (level 0) of the fifth-ordered chromatic scale.	79
5.3	Final chord estimation process.	80
5.4	Key detection results for different pitch ranges.	85
5.5	Evaluation of different key detection algorithms.	88
5.6	Key detection performance for unseen data.	89
5.7	Segmentation of a chromagram.	91
5.8	Diatonic subsets of a chromatic scale.	91
5.9	Diatonic scale visualization of J. S. Bach's Sinfonia No. 3, BWV 789.	93
5.10	Diatonic scale visualization of G. P. da Palestrina's "Missa Papae Marcelli."	94

5.11	Diatonic scale visualization of O. di Lasso’s “Prophetiae Sibyllarum.”	94
5.12	Diatonic scale visualization of a Choral from J. S. Bach’s “Johannespassion” BWV 245.	95
5.13	Diatonic scale visualization of a sonata by L. van Beethoven.	95
5.14	Diatonic scale visualization of R. Wagner’s “Meistersinger von Nürnberg.”	96
5.15	Scale type visualization of C. Debussy’s “Voiles.”	97
5.16	Scale type visualization of C. Debussy’s “La Mer.”	98
5.17	Scale type visualization of I. Stravinsky’s “Le Sacre du Printemps.”	99
5.18	Visualization of O. Messiaen’s modes.	100
6.1	Template-based features for the “Fidelio” orchestra recording.	107
6.2	Interval features for the “Fidelio” example based on different chroma types.	109
6.3	Interval features for the “Fidelio” example in different temporal resolutions.	110
6.4	Linear fit to descending chroma values.	114
6.5	Circular interpretation of chroma vectors.	115
6.6	Example for a scape plot visualization.	116
6.7	Complexity feature values for different tonal items.	117
6.8	Tonal complexity analysis for selected movements from Beethoven’s sonatas.	119
7.1	Overview of the composers in the combined dataset.	123
7.2	Example distribution of a composers works over the lifetime.	125
7.3	Average number of works per year for the different eras.	125
7.4	Estimation of root note progressions.	126
7.5	Relative frequency of root note progressions.	127
7.6	Ratio between authentic and plagal chord progressions distributed over the years.	128
7.7	Root note progressions for different chord types.	128
7.8	Root note progressions of a dominant seventh chord.	129
7.9	Chord types distributed over the years.	130
7.10	Interval type features distributed over the years.	131
7.11	Complexity features distributed over the years.	131
7.12	Self-similarity matrix of root note progressions.	132
7.13	First three principal components from eleven root progression types.	133
7.14	K-means clustering for root note progressions.	134
7.15	Clustering of years for root note progressions.	135
7.16	First three principal components from interval and complexity features.	135
7.17	Self-similarity matrix of interval and complexity features.	136
7.18	K-means clustering based on interval and complexity features.	137
7.19	Clustering of years for local chroma-based features.	137
7.20	Self-similarity matrix based on the feature combination.	138
7.21	Clustering result for a combination of features.	138
7.22	K-means clustering of individual pieces distributed over the years.	140
7.23	K-means clustering of individual pieces as bar histogram.	140
7.24	K-means clustering of composers.	142
7.25	Hierarchical clustering of composers.	143
7.26	Hierarchical clustering using the minimum evolution criterion.	144
8.1	LDA visualizations of the Cross-Era-full dataset.	151
8.2	LDA visualization of the Cross-Era subsets.	153
8.3	LDA visualization of two Cross-Composer subsets.	154
8.4	Schematic overview of the classification procedure.	155
8.5	Classification results for varying model complexity.	161
8.6	Classification accuracy for different temporal resolutions.	162
8.7	Classification results for varying number of LDA dimensions.	165
8.8	Confusion matrices for the individual datasets.	167
A.1	Root note progressions for the individual composers.	182
A.2	Average tonal complexity values for individual composers.	183

List of Tables

2.1	Solfège syllables for the scale degrees of the diatonic modes.	19
2.2	Categorization of root note progressions.	27
2.3	Interval categories and prototypes of pitch class sets.	31
3.1	Different methods for extracting chroma features from audio.	53
4.1	Clustering and classification experiments for musical styles.	74
5.1	Contents of the dataset <i>Symph.</i>	82
5.2	Properties of the key evaluation datasets.	83
5.3	Correct full key classification results for different parameter sets.	84
5.4	Key extraction results for the optimal parameter combination.	86
5.5	Results by historical period.	86
5.6	Results of the MIRtoolbox key detection algorithm.	87
6.1	Chroma feature types for different time scales.	105
6.2	Interval categories.	105
7.1	Cross-Era dataset.	122
7.2	Principal component weights for root note progressions.	134
7.3	Principal component weights for interval and complexity features.	136
8.1	Classification datasets and their properties.	149
8.2	Cross-Composer dataset.	149
8.3	Classification results for different classifiers and datasets.	157
8.4	Classification results with filtering.	159
8.5	Classification experiments for different feature types.	163
8.6	Classification results of a GMM classifier.	166
8.7	Examples for consistently misclassified instances.	169
8.8	Era classification for unseen data.	170
A.1	Dictionary file for the Chordino algorithm.	181

List of Abbreviations

ASE	Audio Spectral Envelope
CEBS	Chroma Estimation using Block Sparsity
CENS	Chroma Energy Normalized Statistics
CLP	Chroma Log Pitch
CP	Chroma Pitch
CQT	Constant-Q Transform
CRP	Chroma DCT-Reduced Log Pitch
CV	Cross Validation
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
EPCP	Enhanced Pitch Class Profile
FFT	Fast Fourier Transform
Fraunhofer IDMT	Fraunhofer Institute for Digital Media Technology
FTC	Fourier-transformed Chroma
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
HPCP	Harmonic Pitch Class Profiles
Hz	Hertz
IF	Instantaneous Frequency
IFC	Instantaneous Frequency Chroma
IRMFSP	Inertia Ratio Maximization using Feature Space Projection
ISMIR	International Society for Music Information Retrieval
KNN	K Nearest Neighbour (Classifier)
LDA	Linear Discriminant Analysis
MFCC	Mel-Frequency Cepstral Coefficient(s)
MIDI	Musical Instrument Digital Interface
MIR	Music Information Retrieval
MIREX	Music Information Retrieval Evaluation eXchange
ML	Machine Learning
MPEG	Moving Picture Experts Group
NN	Neural Networks
NNLS	Non-negative Least Squares
OSC	Octave Spectral Contrast
PCA	Principal Component Analysis
PCP	Pitch Class Profiles
RBF	Radial Basis Function
RC	Reassigned Chroma
RF	Random Forest
SC	Spectral Centroid
SCM	Spectral Crest Measure
SFM	Spectral Flatness Measure
STFT	Short-Time Fourier Transform
SVM	Support Vector Machine
ZCR	Zero-Crossing Rate