A Cross-Version Approach for Harmonic Analysis of Music Recordings

Verena Konz and Meinard Müller*

Saarland University and MPI Informatik
Campus E1-4, 66123 Saarbrücken, Germany
vkonz@mpi-inf.mpg.de, meinard@mpi-inf.mpg.de

Abstract

The automated extraction of chord labels from audio recordings is a central task in music information retrieval. Here, the chord labeling is typically performed on a specific audio version of a piece of music, produced under certain recording conditions, played on specific instruments and characterized by individual styles of the musicians. As a consequence, the obtained chord labeling results are strongly influenced by version-dependent characteristics. In this chapter, we show that analyzing the harmonic properties of several audio versions synchronously stabilizes the chord labeling result in the sense that inconsistencies indicate version-dependent characteristics, whereas consistencies across several versions indicate harmonically stable passages in the piece of music. In particular, we show that consistently labeled passages often correspond to correctly labeled passages. Our experiments show that the cross-version labeling procedure significantly increases the precision of the result while keeping the recall at a relatively high level. Furthermore, we introduce a powerful visualization which reveals the harmonically stable passages on a musical time axis specified in bars. Finally, we demonstrate how this visualization facilitates a better understanding of classification errors and may be used by music experts as a helpful tool for exploring harmonic structures.

1998 ACM Subject Classification H.5.5 Sound and Music Computing, J.5 Arts and Humanities–Music, H.5.1 Multimedia Information Systems, I.5 Pattern Recognition

Keywords and phrases Harmonic analysis, chord labeling, audio, music, music synchronization, audio alignment

Digital Object Identifier 10.4230/DFU.Vol3.11041.53

1 Introduction

Automated chord labeling, which deals with the computer-based harmonic analysis of audio recordings, is one of the central tasks in the field of music information retrieval (MIR) [2, 3, 4, 6, 8, 11, 13, 14, 19, 20, 22, 23]. Harmony is a fundamental attribute of Western tonal music and the succession of chords over time often forms the basis of a piece of music. Thus, chord progressions constitute a powerful mid-level representation for the underlying musical signal and can be applied for various MIR tasks.

* This work has been supported by the Cluster of Excellence on Multimodal Computing and Interaction (MMCI). Meinard Müller is now with Bonn University, Department of Computer Science III, Germany.

Multimodal Music Processing. *Dagstuhl Follow-Ups, Vol. 3.* ISBN 978-3-939897-37-8.
Editors: Meinard Müller, Masataka Goto, and Markus Schedl; pp. 53–72

Dagstuhl Publishing
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Germany

The evaluation of chord labeling procedures is typically performed on large audio collections, where the automatically extracted chord labels are compared to manually generated ground truth annotations. Here, a piece to be analyzed is typically represented by an audio recording, which possesses version-dependent characteristics. For example, specific instruments are used, which have instrument-dependent sound properties, e. g., concerning the energy distributions in the harmonics. Similarly, room acoustics and other recording conditions may have a significant impact on the audio signal's spectral properties. Finally, by emphasizing certain voices or suppressing others, a musician can change the sound in order to shape the piece of music. As a consequence, the chord labeling results strongly depend on specific characteristics of the considered audio recording. Another major problem arises from the fact, that audio-based recognition results refer to the physical time axis given in seconds of the considered audio recording, whereas score-based analysis results obtained by music experts typically refer to a musical time axis given in bars. This simple fact alone makes it often difficult to get musicologists involved into the evaluation process of audio-based music analysis. For example, for the evaluation of chord labeling procedures, ground truth annotations are required. While the manual generation of audio-based annotations is a tedious and time-consuming process musicians are trained to derive chord labels by means of printed sheet music. Such labels, however, are only of limited use for the evaluation of audio-based recognition results. First research efforts have been directed towards the use of score-based ground truth labels for audio-based chord recognition, where it turned out that incorporating such ground truth labels may significantly improve machine learning methods for chord recognition [12, 15].

In this chapter, we build upon a cross-version chord recognition approach previously suggested in [10]. By exploiting the fact that for a musical work there often exist a large number of different audio recordings as well as symbolic representations, we analyze the available versions independently using some automated chord labeling procedure and employ a late-fusion approach to merge the version-dependent analysis results. Here, the idea is to overcome the strong dependency of chord labeling results on a specific version. We show that by using such a cross-version approach one can achieve a stabilization of the chord labeling results. In particular, we observe that more or less random decisions in the automated chord labeling typically differ across several versions. Such passages often correspond to harmonically instable passages leading to inconsistencies. In contrast, consistencies across several versions typically indicate harmonically stable passages. As one main contribution, we show that consistently labeled passages often correspond to correct labeling results. Consequently, one can exploit the consistency information to significantly increase the precision of the result while keeping the recall at a relatively high level, which can be regarded as a stabilization of the labeling procedure. Furthermore, we show that our cross-version approach is conceptually different to a constraint-based approach, where only chord labels are considered that are particularly close to a given chord model. Unlike our cross-version approach, using such simple constraints leads to a significant loss in recall. As another contribution, we describe how to transform the time axis of analysis results obtained from audio recordings to a common musical time axis given in bars. This not only facilitates a convenient evaluation by a musicologist, but also allows for comparing analysis results across different recorded performances.

Finally, we introduce a powerful visualization which is based on the cross-version chord labeling (another interesting approach for visualizing harmonic structures of tonal music has been suggested in [21]). The cross-version visualization indicates the harmonically stable

passages in an intuitive and non-technical way leading the user to passages dominated by a certain key also referred to as tonal centers. Furthermore, in the case that score-based ground truth labels are also provided, the visualization allows for an in-depth error analysis of chord labeling procedures, which deepens the understanding not only for the employed chord recognizer but also for the music material. Additionally, we exemplarily show how the cross-version visualization may serve musicologists as a helpful tool for exploring harmonic structures of a piece of music.

The remainder of this chapter is organized as follows. First, in Section 2 we give an overview of the cross-version chord labeling framework. In Section 3 we show that using a cross-version approach a stabilization of the chord labeling results can be achieved. Afterwards, in Section 4 we exemplarily demonstrate how the cross-version visualization may be used as a supportive tool for exploring harmonic structures before concluding in Section 5 with open problems and future work.

2 Cross-Version Framework

In this section, we describe the cross-version chord labeling procedure following a similar approach as introduced in [10]. Figure 1 shows the employed procedure in a schematic overview. At this point, we emphasize that our approach is not meant to be of technical nature, and we refer to [3, 13] for an overview of state-of-the-art chord labeling procedures. Instead, we introduce a simple yet powerful paradigm which exploits the availability of different versions of a given piece of music.

In the following, we first give a short introduction to music synchronization and describe how synchronization procedures can be used to transform the time axis of audio-based analysis results to a performance-independent musical time axis. Afterwards, we present the employed chord labeling procedure before introducing the concept of cross-version chord labeling. Finally, by means of several music examples, we illustrate the usefulness of our cross-version visualization.

2.1 Synchronization

In the context of the presented cross-version chord labeling approach the concept of music synchronization is of particular importance. In general, the goal of music synchronization is to determine for a given region in one version of a piece of music the corresponding region within another version [9, 17]. Most synchronization algorithms rely on some variant of dynamic time warping (DTW) and can be summarized as follows. First, the two given versions of a piece of music are converted into feature sequences, say $X := (x_1, x_2, \ldots, x_N)$ and $Y := (y_1, y_2, \ldots, y_M)$, respectively. In the synchronization context, chroma features[1] have turned out to yield robust mid-level representations even in the presence of significant musical variations [1, 7, 17, 18]. Chroma features show a high degree of invariance towards changes in timbre and instrumentation while closely correlating to the harmonic progression of the piece of music. From the feature sequences, an $N \times M$ cost matrix C is built up

[1] Implementations of various chroma feature variants are available at www.mpi-inf.mpg.de/resources/MIR/chromatoolbox/, see also [18].

by evaluating a local cost measure c for each pair of features, i. e., $C(n, m) = c(x_n, y_m)$ for $n \in [1 : N] := \{1, 2, \ldots, N\}$ and $m \in [1 : M]$. Then, a cost-minimizing alignment path, which constitutes the final synchronization result, is computed from C via dynamic programming. For a detailed account on DTW and music synchronization we refer to [9, 17] and the references therein. Based on this general strategy, we employ a multiscale synchronization algorithm based on high-resolution audio features as described in [5]. This approach, which combines the high temporal accuracy of onset features with the robustness of chroma features, generally yields robust music alignments of high temporal accuracy.

2.2 Musical Time Axis

The alignment techniques can be used to transform the time axis of audio-based analysis results to a common musical time axis, see Figure 1 for an overview. To this end, we assume that for a certain piece of music we are given a MIDI representation of the musical score, where the MIDI time axis follows a musically meaningful time axis in bars. Such a MIDI file can be obtained by automatically exporting a score in computer-readable format, which in turn can be generated by applying OMR (optical music recognition) software to scanned sheet music, see Figure 1a. Now, given an audio recording of the same piece of music, one can apply music synchronization procedures to establish temporal links between the timelines of the MIDI representation and the audio version. This linking information allows for transferring bar or beat positions from the MIDI timeline to corresponding time positions (given in seconds) of the audio timeline. Then, the audio timeline can be partitioned into segments each corresponding to e. g. one musical beat or bar. Based on this musically meaningful segmentation, beat- or bar-synchronous audio features can be determined. Then each feature vector corresponds to a musically meaningful time unit that is independent of the respective recorded performance. We will use such synchronized features to directly compare the chord labeling results across the different versions.

2.3 Chord Labeling

The chord labeling is then performed on the basis of the synchronized chroma features, where we furthermore apply a tuning estimation to balance out possible deviations of the performances from standard tuning [7, 13]. Note that numerous chord labeling procedures have been described in the literature. State-of-the-art chord recognizers typically employ statistical models such as hidden Markov models [11, 22, 23] or more general graphical models [13] to incorporate smoothness priors and temporal continuity into the recognition process. Since the respective chord labeling procedure is not in the focus of this chapter, we use a basic template-based chord labeling procedure [6], which better illustrates the kind of information that is enhanced and stabilized by our cross-version strategy. However, note that more complex chord recognizers can be used instead.

In the following, we consider 24 chord categories comprising the twelve major and the twelve minor chords, following the conventions as used for MIREX 2010 [16]. Let Λ denote the set of these 24 categories, then for each $\lambda \in \Lambda$ we define a binary template $\mathbf{t}_\lambda$ that corresponds to the respective chord. The template-based chord labeling procedure consists in assigning to each frame (here, exemplarily, we use a bar-wise frame level) the chord label that minimizes a predefined distance d (in our implementation, we use the cosine distance)

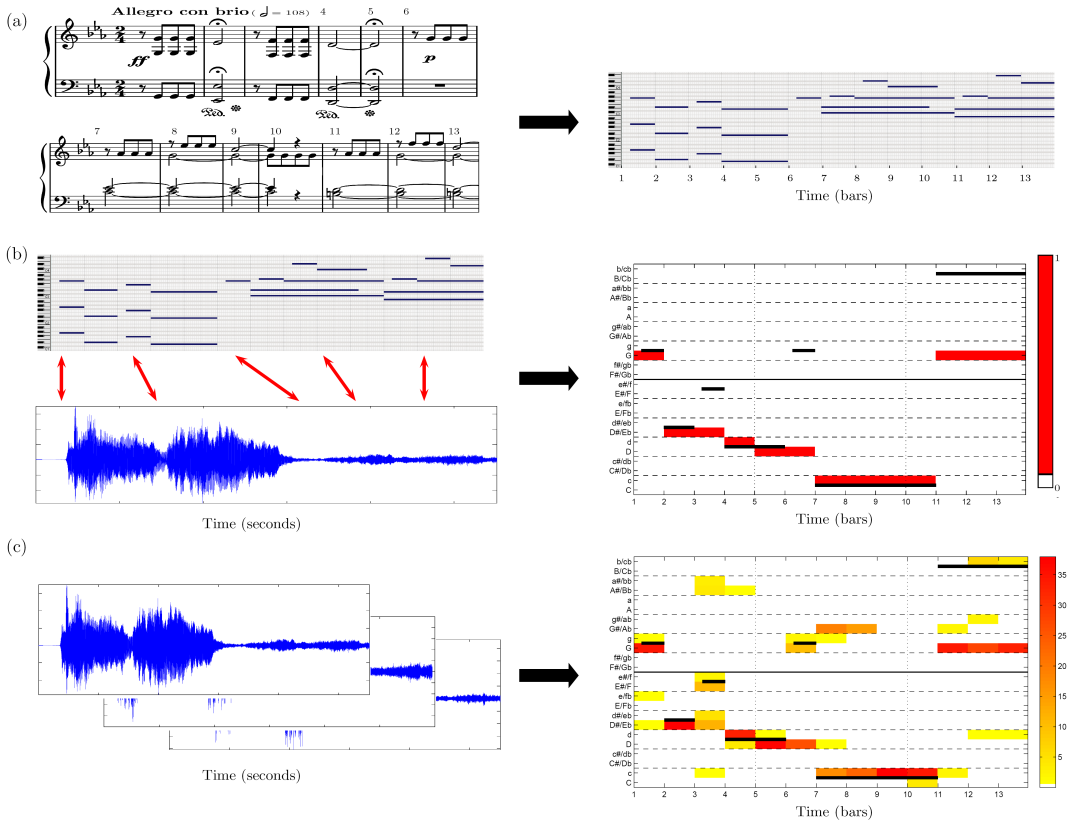

Figure 1 Schematic overview of the employed cross-version framework. Here, the beginning of Beethoven's Fifth (bb. 1-13) is used as an example. (a) Export of the score to a neutral MIDI representation. Here, the score corresponds to a piano reduction of Beethoven's Fifth. (b) Visualization of the automatically derived chord labels for a specific audio recording. The time axis in bars is obtained by synchronizing the audio recording with the MIDI representation. The horizontal black lines in the visualization represent the bassline extracted from the MIDI representation. (c) Cross-version visualization (38 different audio recordings). The horizontal black lines in the visualization represent the bassline extracted from the MIDI representation.

between the corresponding template and a given feature vector referred to as x:

$$\lambda_x := \underset{\lambda \in \Lambda}{\operatorname{argmin}}\ d(\mathbf{t}_\lambda, x). \tag{1}$$

As result, we obtain for each audio version a sequence of automatically extracted bar-wise chord labels. Figure 1b shows the automatically extracted chord labels for a specific audio recording of the first 13 bars of Beethoven's Symphony No. 5, Op. 67, the so-called Beethoven's Fifth. The vertical axis represents the 24 chord categories, where major and minor chords with the same root note are visualized next to each other. Capital letters correspond to major chords, whereas lower case letters correspond to minor chords. The horizontal axis represents the time axis given in bars. The automatically derived chord labels are shown in red, e. g., the chord label for bar 1 corresponds to `G` major, whereas the chord label for bar 2 corresponds to `E♭` major. As the bassline of a harmonic progression plays an important role for the understanding of harmonic structures, we have visualized it as an additional information in the middle of the corresponding major and minor chord having the bassline

as root note. The bassline is automatically extracted from the MIDI representation by determining the lowest of all present MIDI notes at every point in time.

2.4 Cross-Version Chord Labeling

As mentioned in the introduction, the chord labeling results not only depend on the piece of music but also on the acoustic and artistic characteristics of the specific audio recording. To alleviate the dependence on such characteristics, one can exploit the fact that for classical pieces of music usually many different recorded performances exist. Here, our idea is to perform the chord labeling across several versions of a given piece of music and then to resolve the dependency of the chord labels on a specific version by using some kind of late-fusion strategy. Since the automatically extracted chord labels for the different performances are given bar-wise, one can overlay the performance-specific chord labels for all considered recorded performances resulting in a cross-version visualization. Figure 1c shows a cross-version visualization for the beginning of Beethoven's Fifth (bb. 1-13), where 38 different performances are considered. The color-scale ranging from bright yellow to dark red indicates the degree of consistency of the chord labels across the various performances, where red entries point to consistencies and yellow entries to inconsistencies. For example, bar 2 is labeled highly consistently, whereas bar 3 is labeled inconsistently across the considered performances.

In this way, the cross-version visualization directly reveals chord label consistencies and inconsistencies across the different performances giving a deeper insight into the chord labeling procedure as well as the underlying music material. As we will show, consistently labeled passages generally correspond to harmonically stable passages, which are clearly dominated by a certain key. In some cases, consistencies may also point to consistent misclassifications which might be taken as an indicator for inadequacies of the underlying chord labeling model. For example, considering only 24 major and minor chords, it is obvious that more complex chords such as, e. g., diminished chords can not be captured. In contrast, inconsistencies generally point to harmonically instable passages or ambiguities in the underlying music material. For example, incomplete chords as well as additional notes such as trills, appoggiaturas or suspended notes lead to chord ambiguities causing an inconsistent labeling across the different performances.

2.5 Examples

To illustrate our cross-version approach, we now discuss some real-world music examples. We first refer to the introductory bars of Beethoven's Fifth (see Figure 1). Figure 1b shows the visualization of the automatically derived chord labels for a specific audio recording. Following the time axis in bars, the visualization allows for a direct comparison to the score. As the score reveals the first five bars (bb. 1-5) do not contain complete triads. Instead, the characteristic "fate motif" appears, which is presented in octaves in unison. The visualization shows that the automatically derived chord labels for these introductory bars, aside from bar 3, are meaningful in the sense that they represent chords having the presented note of the respective bar as root note. However, in bar 3, where `f` is played in unison, `E♭` major is detected. This might be an indicator for inaccuracies in the synchronization since the previous bar (b. 2) is dominated by the note `e♭`. The same problem appears in bar 6. Bars 7-10 are then labeled as `C` minor. A closer look at the score reveals that in this passage

(bb. 8-10) C minor is clearly present. However, in the beginning of this passage (b. 7) C minor with suspended sixth (a♭) leads into the C minor chord (bb. 8-10). In fact, C minor with suspended sixth corresponds to the notes of A♭ major. However, the suspended sixth (a♭) is played in a very soft way in the considered recording, which might be the reason for the detection of C minor. Bars 11-13 then are labeled in a meaningful way as G major.

The cross-version visualization (Figure 1c) now directly reveals consistently and inconsistently labeled passages. For example, one observes the following highly consistently labeled passages, which may correspond to harmonically stable passages: bars 1-2, 4-5 and 8-13. As previously described, bars 1-2 and 4-5 refer to the fate motif in unison, thus not containing complete triads. These bars are now consistently labeled as a chord having the respective note of the considered bar as root note. Comparing bars 8-13 to the score shows that they indeed correspond to passages being clearly dominated by a certain harmony. Bars 8-10 are consistently labeled correctly as C minor reflecting the harmonic stability of this passage, which is clearly dominated by a C minor triad. Similarly, bars 11-13 are correctly identified by the visualization as harmonically stable, being dominated by G major. In contrast, one directly observes that bar 3 is labeled inconsistently. This inconsistent labeling may be due to local inaccuracies in the underlying synchronization procedure. For a larger amount of recordings this bar is labeled as F major (or as E♭ major) having as root the note presented in unison in this bar (or in the previous bar). In fact, bar 3 was already misclassified as E♭ major considering a single audio recording before. The cross-version visualization now clearly identifies this bar to be problematic in view of the underlying synchronization procedure. Finally, bar 7 attracts attention since it is labeled for approximately half of the recordings as C minor and as A♭ major for the other half. Here, C minor with suspended sixth (a♭) is present, which indeed sounds equivalently to A♭ major. Since the suspended a♭ is usually played in a soft way, for many recordings (including the previously discussed specific recording) this bar is misclassifed as C minor. However, the cross-version visualization shows that for the largest part of recordings this bar is correctly classified (with regard to the sound) as A♭ major.

As the previously discussed example shows, ground truth data is not necessarily needed to derive valuable information from the cross-version visualization concerning the employed chord labeling procedure as well as the underlying music material. However, assuming the case that score-based ground truth labels are provided by a trained musician, this information can be easily incorporated into our cross-version approach, see Figure 2. In this way, errors (deviations from the ground truth) can be subdivided into errors being specific to a certain audio version (inconsistent misclassifications) and errors independent of a specific version (consistent misclassifications). While inconsistent misclassifications may point to ambiguities in the underlying music material, consistent misclassifications may point to inadequacies in the underlying chord labeling framework. In the following, we illustrate such an in-depth error analysis by means of two examples. The score-based ground truth annotations used in our experiments have been generated by a trained musician on the bar-level using the shorthands and conventions proposed by Harte et al. [8].

Figure 2 shows the cross-version visualization for a different excerpt of Beethoven's Fifth (bb. 40-47). On the left, the previously introduced visualization is shown, where the automatically derived cross-version chord labels are visualized without considering ground truth chord labels. On the right, an extension of this cross-version visualization is presented, where the cross-version chord labels are compared to score-based ground truth labels. In this visualization we now distinguish two different color scales: one color scale ranging from dark blue to bright green and the previously introduced color scale ranging from dark red

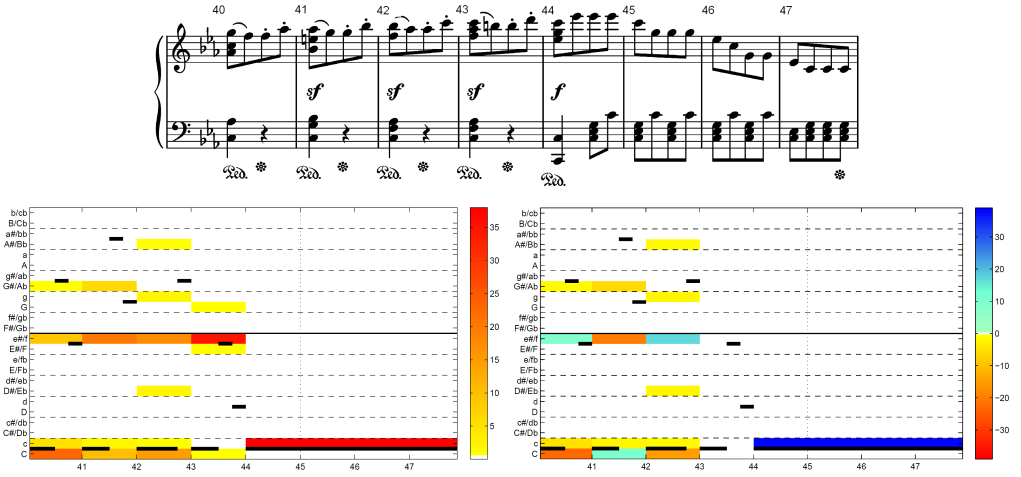

Figure 2 Cross-version visualization for Beethoven's Fifth (bb. 40-47). Here, 38 different audio recordings are considered. **Left:** Cross-version visualization of the automatically derived chord labels. **Right:** Cross-version visualization, where the automatically derived chord labels are overlayed with score-based ground truth chord labels.

to yellow. The first color scale from blue to green serves two purposes. Firstly, it encodes the score-based ground truth chord labels. Secondly, it shows the degree of consistency between the automatically generated audio labels and the score labels. For example, the dark blue entries in bars 44-47 show, that a `C` minor chord is specified in the score-based ground truth labels, and all automatically derived chord labels coincide with the score label here. In contrast, the bright green entry in bar 40 shows that the score-based chord label corresponds to `F` minor, but most of the automatically derived chord labels differ from the score label, specifying a `C` major chord. Analogously, the second color scale from dark red to yellow also fulfills two purposes. Firstly, it encodes the automatically derived chord labels that differ from the score-based labels. Secondly, it measures the universality of an error. For example, in bars 44-47 there are no red or yellow entries, since the score-based labels and the automatically derived labels coincide here. However, in bar 40 most automatically derived chord labels differ from the score-based labels. Here most chord labels specify a `C` major chord.

The cross-version visualization of the automatically derived chord labels (see Figure 2, left) reveals two highly consistently labeled passages: bar 43, labeled highly consistently as `F` minor, and bars 44-47, which are labeled as `C` minor across all considered recorded performances. Comparing to the score, bars 44-47 indeed turn out to be a harmonically stable passage which is clearly dominated by `C` minor. Consequently, this highly consistently labeled passage is labeled correctly, which is shown in the visualization, where the automatically derived chord labels are compared to score-based ground truth labels (see Figure 2, right). In contrast, bar 43 is labeled consistently as `F` minor (see Figure 2, left), but comparing to the score one finds out that besides of an `F` minor chord two additional notes (`b` and `d`) are contained in this bar, suggesting the dominant `G` major. Therefore, a clear assignment of a triad is not possible on the bar level. This is also the reason that there is no score-based label assigned to this bar in the ground truth annotation (see Figure 2, right). The remaining bars are labeled rather inconsistently indicating harmonic instability or ambiguities in the underlying music material (see Figure 2, left). A closer look at the score reveals that these

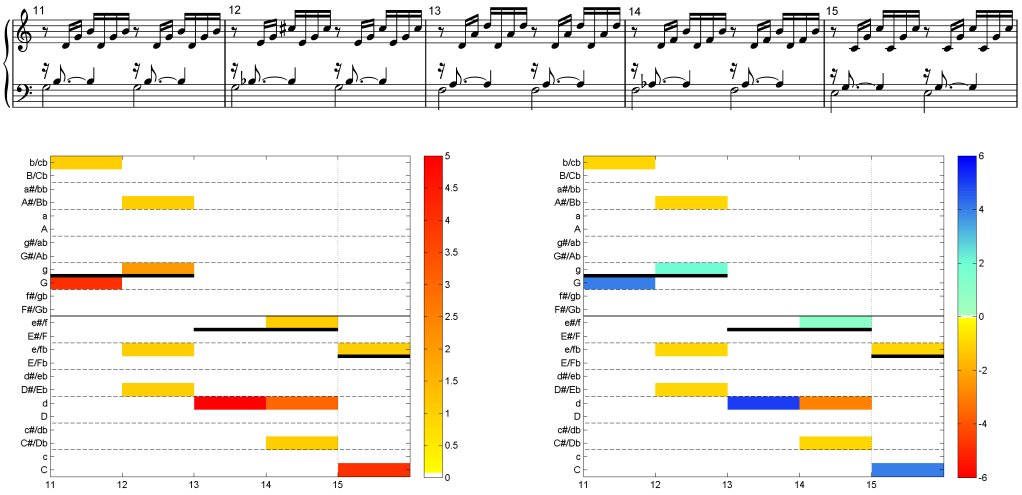

Figure 3 Cross-version visualization for Bach's Prelude BWV 846 in C major (bb. 11-15). Here, five different audio recordings are considered. **Left:** Cross-version visualization of the automatically derived chord labels. **Right:** Cross-version visualization, where the automatically derived chord labels are overlayed with score-based ground truth chord labels.

bars are characterized by suspended notes on the first beat. These additional notes which do not belong to the underlying chords are mainly responsible for the inconsistent labeling. The comparison with the score-based ground truth annotation reveals that for bars 40 and 41 indeed most of the automatically derived chord labels differ from the ground truth annotation (see Figure 2, right).

Figure 3 shows the cross-version visualization for an excerpt of Bach's Prelude BWV 846 in C major (bb. 11-15), where five different recorded performances are considered. The visualization reveals 3 bars which are labeled correctly with high consistency (b. 11, b. 13, and b. 15) and two bars, which are misclassified for most of the considered audio versions (b. 12 and b. 14). Comparing to the score one finds out that the correctly labeled passages indeed correspond to bars, where clear major or minor chords are present. In contrast, bars 12 and 14 are problematic in the sense that they contain diminished seventh chords which can not be assigned in a meaningful way to one of the considered 24 major and minor chords, thus producing misclassifications. In this case, an extension of the considered chord categories to also include diminished seventh chords might solve the problem.

3 Stabilizing Chord Labeling

In this section we show that analyzing the harmonic properties of several audio versions synchronously stabilizes the chord labeling result in the sense that inconsistencies indicate version-dependent characteristics, whereas consistencies across several versions indicate harmonically stable passages in the piece of music. To this end, we introduce a cross-version voting strategy and compare it with a simple constraint-based strategy using a single version. This comparison demonstrates that our voting strategy is conceptually different from simply imposing stricter conditions in the template-based approach. The two strategies are illustrated by means of the first 19 bars of Beethoven's Piano Sonata Op. 27 No. 2, the so-called Moonlight Sonata, see Figure 4.

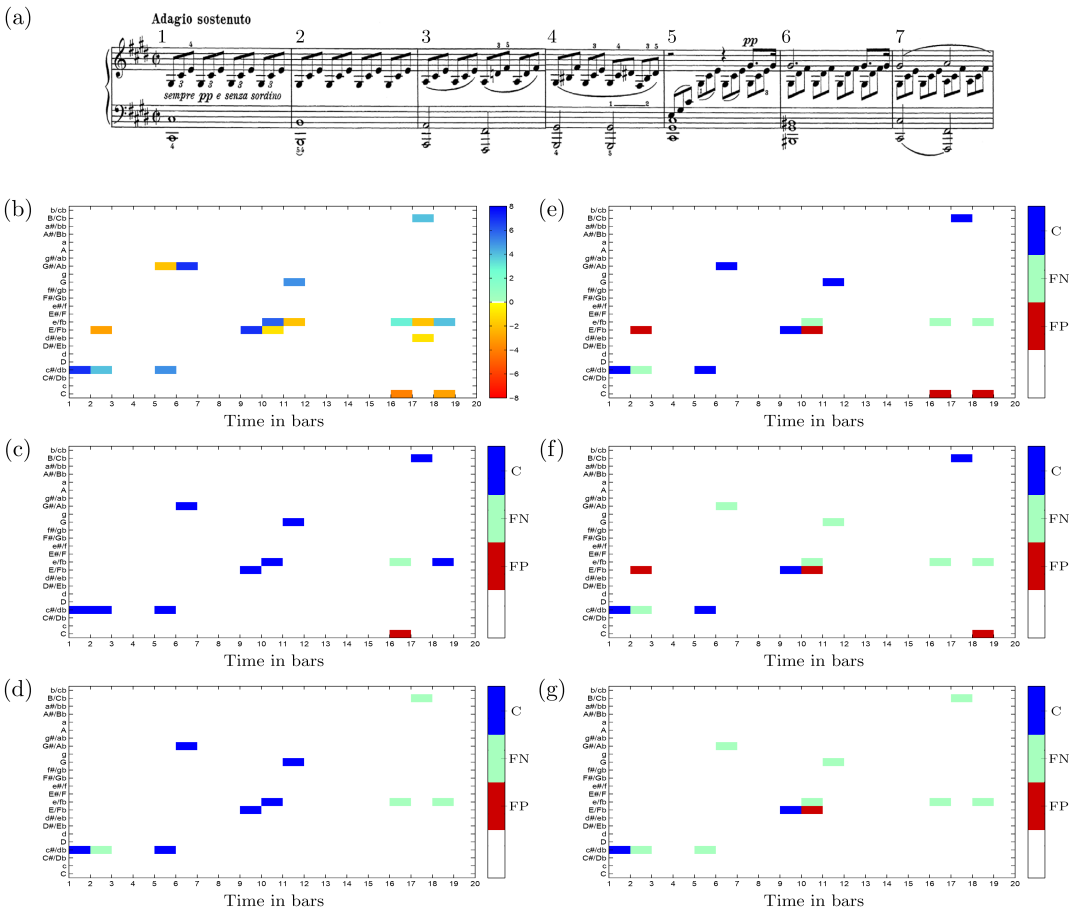

Figure 4 Visualization of the chord labeling result for Beethoven's Moonlight Sonata (bb. 1-19). In the left column (b-d) the cross-version voting strategy is used considering seven performances, whereas in the right column (e-g) the constraint-based strategy is used considering only a single audio recording (Barenboim). Bars, for which no score-based ground truth label exists (since the clear assignment of a harmony is not possible), are left unconsidered in the evaluation. (a) Score of bars 1-7. (b) Visualization of consistencies and inconsistencies in the cross-version analysis. (c) Cross-version majority voting strategy. (d) Cross-version voting strategy with $\nu = 0.5$. (e) Basic strategy. (f) Constraint-based strategy with $\gamma = 0.3$. (g) Constraint-based strategy with $\gamma = 0.1$.

3.1 Cross-Version Voting Strategy

By overlaying the chord labeling results as described in Section 2.4 for the first 19 bars of Beethoven's Moonlight Sonata considering seven different audio versions, we obtain a cross-version visualization, see Figure 4b. The cross-version strategy now reveals consistencies and inconsistencies in the chord labeling across all audio versions. For example, one directly notices that the misclassification in bar 10, when considering a specific audio version (see Figure 4e), seems to be version-dependent. Considering several audio versions, bar 10 is more or less consistently labeled correctly as `E` minor. In contrast, a more consistent misclassification (`C` major instead of `E` minor was labeled for four versions) can be found in bar 16.

In the following experiment, we investigate to which extent the consistency information across several audio versions may be exploited to stabilize chord labeling. In the *majority voting strategy* we keep for each bar exactly one of the automatically extracted chord labels, namely the most consistent chord label across all versions. All remaining audio chord labels are left unconsidered in the evaluation. This results in a visualization which is shown in Figure 4c. Blue entries (correct: C) now indicate areas, where the audio chord label agrees with the ground truth chord label. In contrast, green and red entries encode the differences between the chord labels. Here, red entries (false positives: FP) correspond to the audio chord labels, whereas green entries (false negatives: FN) correspond to the ground truth labels. As one directly notices, besides one misclassification in bar 16, the above mentioned highly consistent error, all chords are now correctly classified resulting in a significant increase of precision.

In the next step, we further constrain the degree of consistency by introducing a consistency parameter $\nu \in [0, 1]$. To this end, we consider only bars which are labeled consistently for more than $(\nu \cdot 100)\%$ of the audio versions. All other bars are left unannotated. For example, $\nu = 0.5$ signifies that we keep in the evaluation only passages, where for more than 50% of the audio versions the extracted chord labels agree. Figure 4d shows the visualization of the chord labeling result for $\nu = 0.5$, where the voting procedure succeeds in eliminating all misclassifications. At the same time only three correct classifications are taken out of the evaluation. In this way, the precision further increases (amounting to 100% in Figure 4d), while the recall still remains on a relatively high level (amounting to 60% in Figure 4d).

As the example described above shows, the cross-version voting approach succeeds in significantly increasing the precision, while keeping the recall at a relatively high level. For a quantitative evaluation of the cross-version voting strategy we refer to the experiments described in Section 3.3.

3.2 Constraint-Based Strategy

To better illustrate the potential of our cross-version voting strategy, we now consider a constraint-based stabilizing procedure. Using the template-based approach described in Section 2.3, the automatically derived chord label for a given bar is defined by the template having the minimal distance to the feature vector, in the following referred to as *basic strategy*. Figure 4e shows a visualization of the chord labeling result. As the visualization reveals the first bar is correctly identified as C♯ minor, whereas bar 2 is misclassified, being identified as E major although being labeled as C♯ minor in the ground truth. Here, a C♯ minor 7th chord is present in the ground truth, being mapped to C♯ minor. In fact, this seventh chord contains all the tones for E major, which explains the misclassification.

As we can see from the example, using the basic strategy, it obviously happens that for bars containing complex chords none of the given 24 templates fits well to the present feature vector. Here, the chord template of minimal distance may have a rather large distance to the feature vector. To counteract this case, we now introduce a parameter $\gamma \in [0, 1]$, which represents an upper threshold for the distance between the assigned chord template and the feature vector. In this way, we obtain a constraint-based procedure, where only chord labels λ are kept for which

$$d(\mathbf{t}_\lambda, x) < \gamma. \tag{2}$$

Table 1 Overview of the pieces and number of versions used in our experiments.

Composer	Piece	# (Versions)	Identifier
Bach	Prelude C Major BWV 846	5	'Bach'
Beethoven	Moonlight Sonata Op. 27 No. 2 (first movement)	7	'BeetM'
Beethoven	Fifth Symphony Op. 67 (first movement)	38	'Beet5'
Chopin	Mazurka Op. 68 No. 3	49	'Chopin'

All feature vectors x that have a larger distance than γ to any of the chord templates are left unannotated. In the following experiment, the idea is to successively decrease the parameter γ in order to investigate its influence on the chord labeling result.

Figure 4f shows the visualization for $\gamma = 0.3$. Obviously, one misclassification (bb. 16) is now taken out of the evaluation. However, at the same time two previously correctly classified chords (bb. 6, bb. 11) are left unconsidered in the evaluation, resulting in a decrease of the recall. Here, again seventh chords are present being correctly classified but having a relatively large distance to the template vector. Further decreasing the parameter γ is accompanied by a dramatical loss in recall while the precision increases moderately (Figure 4g). For quantitative results of the evaluation of the constraint-based strategy we refer to the experiments shown in Figure 5.

3.3 Experiments

In this section we quantitatively evaluate the various chord labeling strategies using a dataset that comprises four classical pieces of music, see Table 1. At this point, we want to emphasize that our main object is not in increasing the F-measure, defined below. Instead, in the application we have in mind, we are interested in finding passages, where one obtains correct chord labels with high guarantee. Therefore, our aim is to increase the precision, however, without losing too much of the recall.

In the following, we denote the automatically derived audio chord labels as L_a, and the ground truth chord labels as L_{gt}. For our bar-wise evaluation, we use precision (P), recall (R) and F-measure (F) defined as follows:

$$P = \frac{\#(L_a \cap L_{gt})}{\#L_a}, \quad R = \frac{\#(L_a \cap L_{gt})}{\#L_{gt}}, \quad F = 2 \cdot \frac{P \cdot R}{P + R}. \tag{3}$$

We first discuss the cross-version voting strategy. Figure 5 shows curves for P, R and F for the four pieces in the dataset, where the horizontal axis now represents the parameter ν ranging between 0.5 and 0.8 except for the position labeled by 'Maj' corresponding to the majority voting strategy. First of all, one notices that performing the chord labeling across several versions using the majority voting strategy, precision, recall and F-measure already improve by 10-30% in comparison to the basic strategy based on a specific version (see 'Min' in Figure 5).

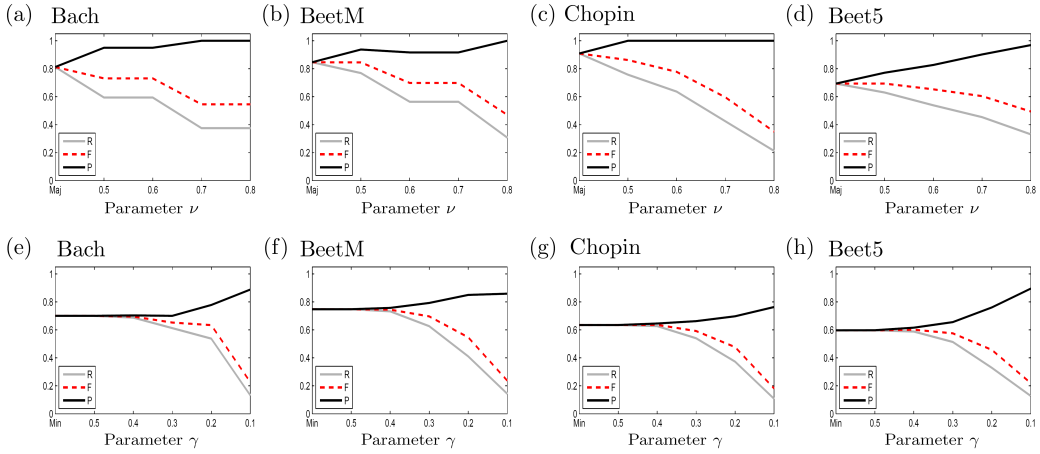

Figure 5 Top: Cross-version voting strategy. Curves for precision (P), recall (R) and F-measure (F) using the majority voting strategy (Maj), and four different consistency parameters ν from 0.5 to 0.8. **Bottom:** Constraint-based strategy based on a specific version. Curves for the mean value of precision (P), recall (R) and F-measure (F) using the basic strategy (Min) and five different settings for γ from 0.5 to 0.1.

Table 2 Basic chord labeling based on specific versions. The table shows mean, minimum and maximum F-measures over all recorded performances of a given piece.

	Mean	**Min**	**Max**
Bach	0.7000	0.4375	0.8750
BeetM	0.7473	0.6923	0.8718
Chopin	0.6345	0.4545	1.0000
Beet5	0.5967	0.5282	0.8345

Furthermore, for all four examples the precision rapidly increases, so that for $\nu = 0.5$ already a high precision is reached: 95% (Bach), 94% (BeetM), 100% (Chopin) and 77% (Beet5). At the same time the recall remains on a rather high level, still amounting to 59% (Bach), 77% (BeetM), 76% (Chopin) and 63% (Beet5). In this way, our experiments show that consistently labeled passages across several versions often correspond to correctly labeled passages. Increasing the consistency parameter ν further increases the precision values, while the recall still remains at acceptably high levels. In summary, exploiting the consistency information of the chord labels across several versions succeeds in stabilizing the chord labeling, resulting in a significant increase of precision without loosing too much of the recall.

We now compare these results with the ones obtained from the constraint-based strategy. Figure 5 shows curves for P, R, and F for the four pieces in our dataset. Here, P, R, and F correspond to mean values, which are obtained by first applying the constraint-based strategy on every version in the dataset separately and then averaging over all these versions. In the visualization, the horizontal axis represents the parameter γ ranging between 0.5 and 0.1 except for the position labeled by 'Min' corresponding to the basic labeling strategy. As one directly notices, there is a clear tendency visible for all four examples in our database. For increasing γ the precision also slowly increases reaching a high value of roughly 80% for

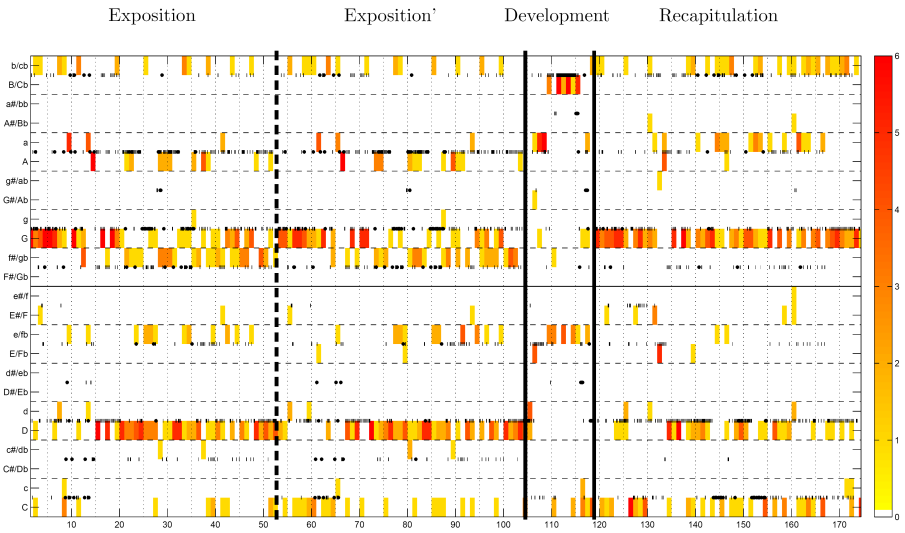

Figure 6 Cross-version visualization for the first movement of Beethoven's Piano Sonata Op. 49 No. 2. Here, six different recorded performances are considered.

$\gamma = 0.1$. However, at the same time the recall dramatically drops down to roughly 10% for $\gamma = 0.1$. Obviously, using the constraint-based strategy one can also increase precision values as misclassifications are taken out of the evaluation, however at the same time previously correct classifications are excluded resulting in a declining recall. Because of the dramatic loss of recall, this simple constraint-based strategy is not suited for stabilizing the chord labeling results.

Furthermore, our experiments reveal that performing the chord labeling based on a specific audio recording, the version-dependent results can vary greatly. This is shown by Table 2 indicating the mean F-measure, as well as the minimal and maximal F-measure achieved over all available recordings when using the basic labeling strategy (there was also a MIDI-synthesized version in each of the four groups). For example, the F-measure for one version of Bach amounts to 43.75%, corresponding to the minimal F-measure over all versions, whereas for another version the F-measure amounts to 87.5%, corresponding to the maximal F-measure over all versions. The average F-measure over the five versions amounts to 70%. These strong variations of the chord labeling results across different versions can not be explained by tuning effects, as we compensated for possible tuning deviations in the feature extraction step. A manual inspection showed that, for most cases, musical ambiguities are responsible for strong differences between the version-dependent results.

4 Exploring Harmonic Structures

As the experiments described above have shown, consistently labeled passages across several versions often correspond to correctly labeled passages. This opens the way for large-scale harmonic analyses on the basis of huge recorded music corpora, where cross-version chord labels of high reliability can be used instead of manually generated ground truth labels. In current work, we apply our cross-version analysis framework for automatically revealing hidden harmonic relations across different pieces of specific music corpora. In particular, in a collaboration with musicologists, we are investigating how to locate tonal centers, i. e.

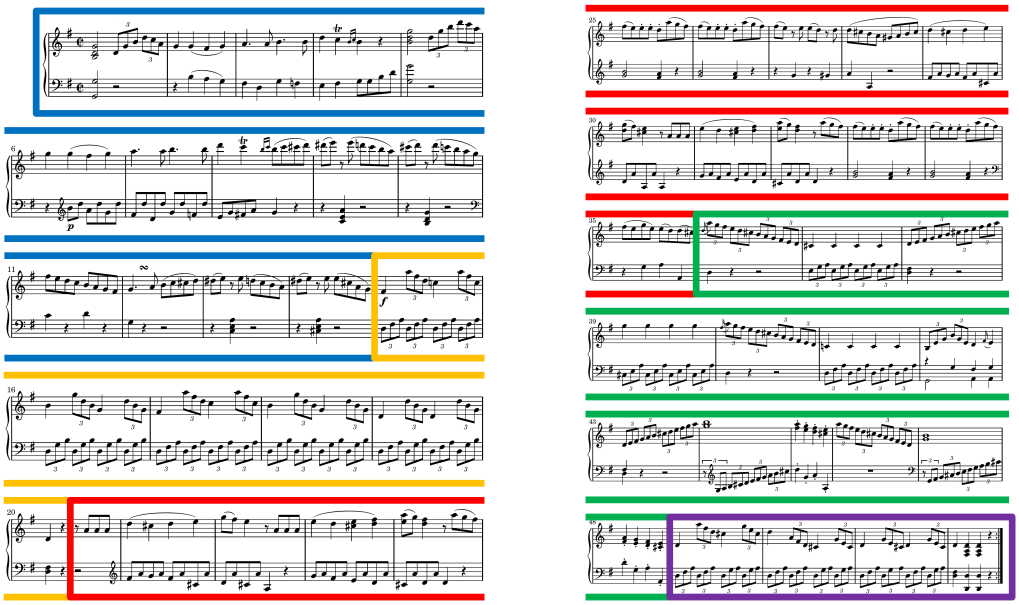

Figure 7 Exposition (bb. 1-52) of Beethoven's Piano Sonata Op. 49 No. 2. Musically meaningful sections are marked in the score: first group (blue), transition (yellow), second group (red), third theme (green), cadential group (purple).

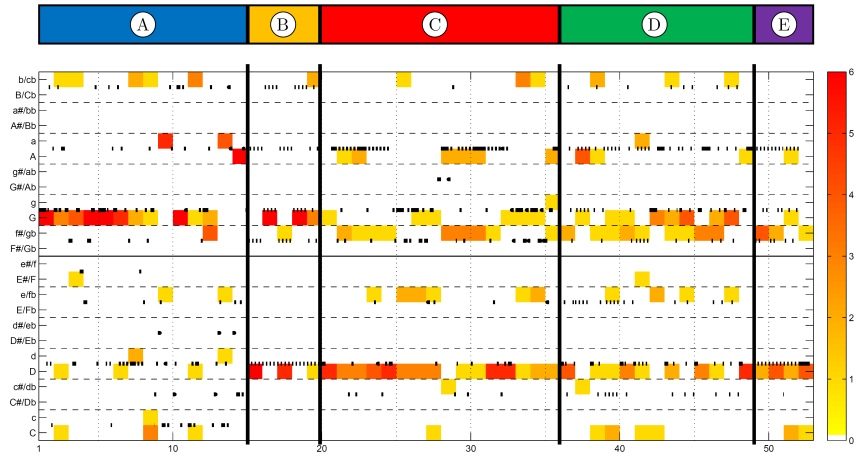

Figure 8 Cross-version visualization for the exposition of Beethoven's Piano Sonata Op. 49 No. 2. Here, six different recorded performances are considered.

passages which are dominated by a certain key, within large music corpora. Here, in the context of a harmonic analysis on a relatively coarse temporal level, our cross-version analysis has turned out to be a valuable tool that can reliably differentiate between harmonically stable and harmonically instable passages.

In the following, we exemplarily demonstrate how our cross-version visualization may serve musicologists as a helpful tool for exploring harmonic structures of a musical work. As example we use the first movement of Beethoven's Sonata Op. 49 No. 2 (see Figure 7). Figure 6 shows the cross-version visualization as an overview, where six different recorded performances are considered. The first movement is divided into three different form

parts: exposition (bb. 1-52) and its repetition (bb. 53-104), development (bb. 105-118) and recapitulation (bb. 119-174).[2] These parts are marked by vertical black lines and can be clearly separated from each other by their harmonic structures. The exposition is clearly dominated by the tonic G major and the dominant D major, which represent the keys of the first and the second theme, respectively. In contrast, the development is characterized by a greater variety of quickly changing harmonies: mainly D minor, A minor, E major, E minor and B major appear in the visualization as tonal centers. Finally, in the recapitulation, the tonic G major is stabilized: it appears now as the main tonal center (the second theme is likewise presented in the tonic), supported by shorter appearances of subdominant C major and dominant D major.

Apart from reflecting large-scale harmonic structures, the cross-version visualization allows for a more detailed bar-wise harmonic analysis, which we now exemplarily perform for the exposition of the sonata (see Figure 7 and Figure 8). The exposition is divided into five musically meaningful subparts, which again are characterized by specific harmonic structures: first group (A; bb. 1-14), transition (B; bb. 15-20), second group (C; bb. 20-35), third theme (D; bb. 36-48) and cadential group (E; bb. 49-52). These subparts of the exposition are marked by vertical black lines and displayed as color-coded blocks on top of the visualization (for a comparison to the score, see Figure 7).

As the visualization reveals the first theme (A) is characterized by harmonic stability, especially in the beginning where the tonic G major is clearly present. However, one directly observes two bars which are labeled inconsistently across the various performances indicating harmonic instability: bars 7 and 8. Comparing to the score, one finds out that in bar 7 indeed two different harmonies appear, which is the reason for the inconsistent labeling on the bar-level. Similarly, bar 8 contains several harmonies including a diminished seventh chord, a chromatic melodic line and a trill so that no unique chord label can be assigned to this bar. The transition (B) is characterized by harmonic stable passages in the tonic G major and the dominant D major. As the score reveals, this section is indeed characterized by a bar-wise change between these two chords so that the transition leads to the entrance of the second theme (C) appearing in the dominant D major. The visualization clearly reflects that the key of the second theme is D major. However, some of the bars also exhibit inconsistencies. For example, bars 21-24 are classified as F♯ minor instead of D major for some of the recordings. A closer look at the score reveals that in these introductory bars of the second theme the leading tone c♯ of D major is often present, which musically stabilizes D major but at the same time produces (together with the notes f♯ and a of the D major chord) a chord ambiguity leading to the classification F♯ minor for some of the performances. A similar confusion occurs in bars 28-30, where the performers strongly emphasize the leading tone c♯. The second theme is followed by a kind of third theme (D), which exhibits many inconsistencies. A comparison to the score shows that this passage is characterized by chromatic runs which are responsible for the inconsistent labeling across the considered performances. The exposition finally closes with the cadential group (E), which usually stabilizes the tonic in the end. Surprisingly, the visualization reveals that the labeling for this section is not as consistent as one may expect. Here, the score shows that the tonic D major indeed dominates this passage, but the leading tone c♯ appears again together with the suspended fourth g.

[2] Note, that bar numbering in printed sheet music usually does not take into account repetitions.

5 Conclusions

In this chapter, we presented a cross-version approach for chord labeling. In particular, we showed that consistently labeled passages across several versions often correspond to correctly labeled passages. Presenting the cross-version analysis results on a musically meaningful time axis in bars also helps to make the analysis results better accessible to music experts. Firstly, the presented approach allows for involving musicologists in the evaluation process of automated chord labeling procedures. For example, the cross-version visualization opens the way for an interdisciplinary collaboration, where musicologists may greatly support computer scientists in performing an in-depth error analysis of the employed chord labeling procedure based on the score. Secondly, the cross-version visualization may serve musicologists as a helpful tool for exploring harmonic structures of a musical work. Because of their high reliability, cross-version chord labels may be an alternative to manually generated ground truth labels. This may particularly hold for large-scale harmonic analyses on the basis of huge corpora of recorded music.

As for future work, we need to perform more detailed quantitative evaluations to verify our hypothesis that our cross-version approach indeed leads to a stabilization of the chord labeling results. Furthermore, we plan to apply our cross-version framework on the entire corpus of Beethoven's piano sonatas. In collaboration with musicologists, we are currently investigating harmonic structures across different movements for some of the sonatas. Here, our automated methods may help to investigate which tonal centers occur in a specific sonata and how they are functionally related to each other. In this context, a structure-oriented analysis, which analyzes tonal centers according to the different form parts of the classical sonata form, is of great musicological meaning as each such part is characterized by a specific occurrence of certain harmonies. Performing this analysis across the complete corpus of Beethoven's piano sonatas, we aim to quantify and better understand from a music-historical perspective how Beethoven has applied tonal centers in his work. Finally, we plan to use our automated framework for exploring harmonic structures across even larger and more complex corpora of musical works, such as the corpus of Wagner's operas. Here, due to the vast amount of data, a purely manual harmonic analysis is hardly possible. Also, being characterized by complex harmonies and rich orchestrations, the detection of large-scale harmonic relations within and across the operas becomes a challenging task.

6 Acknowledgment

We would like to express our gratitude to Michael Clausen, Cynthia Liem, and Matthias Mauch for their helpful and constructive feedback.

References

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

2 Juan Pablo Bello and Jeremy Pickens. A robust mid-level representation for harmonic content in music signals. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 304–311, London, UK, 2005.

3 Taemin Cho, Ron J. Weiss, and Juan Pablo Bello. Exploring common variations in state of the art chord recognition systems. In *Proceedings of the Sound and Music Computing Conference (SMC)*, pages 1–8, Barcelona, Spain, 2010.

4 Ching-Hua Chuan and Elaine Chew. Audio key finding: Considerations in system design and case studies on Chopin's 24 Preludes. *EURASIP Journal on Advances in Signal Processing*, 2007:1–15, 2007.

5 Sebastian Ewert, Meinard Müller, and Peter Grosche. High resolution audio synchronization using chroma onset features. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1869–1872, Taipei, Taiwan, 2009.

6 Takuya Fujishima. Realtime chord recognition of musical sound: A system using common lisp music. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 464–467, Beijing, 1999.

7 Emilia Gómez. Tonal description of polyphonic audio for music content processing. *INFORMS Journal on Computing*, 18(3):294–304, 2006.

8 Christopher Harte, Mark Sandler, Samer Abdallah, and Emilia Gómez. Symbolic representation of musical chords: A proposed syntax for text annotations. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 66–71, London, GB, 2005.

9 Ning Hu, Roger B. Dannenberg, and George Tzanetakis. Polyphonic audio matching and alignment for music retrieval. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, US, October 2003.

10 Verena Konz, Meinard Müller, and Sebastian Ewert. A multi-perspective evaluation framework for chord recognition. In *Proceedings of the 11th International Conference on Music Information Retrieval (ISMIR)*, pages 9–14, Utrecht, The Netherlands, 2010.

11 Kyogu Lee and Malcolm Slaney. A unified system for chord transcription and key extraction using hidden Markov models. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Vienna, AT, 2007.

12 Robert Macrae and Simon Dixon. Guitar tab mining, analysis and ranking. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 453–458, Miami, USA, 2011.

13 Matthias Mauch and Simon Dixon. Simultaneous estimation of chords and musical context from audio. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6):1280–1289, 2010.

14 Matthias Mauch, Daniel Müllensiefen, Simon Dixon, and Geraint Wiggins. Can statistical language models be used for the analysis of harmonic progressions? In *Proceedings of the International Conference of Music Perception and Cognition (ICMPC)*, Sapporo, Japan, 2008.

15 Matt McVicar, Yizhao Ni, Raul Santos-Rodriguez, and Tijl De Bie. Using online chord databases to enhance chord recognition. *Journal of New Music Research*, 40(2):139–152, 2011.

16 MIREX 2010. Audio Chord Estimation Subtask. `http://www.music-ir.org/mirex/wiki/2010:Audio_Chord_Estimation`, Retrieved 17.09.2010.

17 Meinard Müller. *Information Retrieval for Music and Motion*. Springer Verlag, 2007.

18 Meinard Müller and Sebastian Ewert. Chroma Toolbox: MATLAB implementations for extracting variants of chroma-based audio features. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 215–220, Miami, USA, 2011.

19 Hélène Papadopoulos and Geoffroy Peeters. Joint estimation of chords and downbeats from an audio signal. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(1):138–152, 2011.

20 Jeremy T. Reed, Yushi Ueda, Sabato Siniscalchi, Yuki Uchiyama, Shigeki Sagayama, and Chin-Hui Lee. Minimum classification error training to improve isolated chord recognition. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 609–614, Kobe, Japan, 2009.

21 Craig Stuart Sapp. *Computational Methods for the Analysis of Musical Structure*. PhD thesis, Stanford University, USA, May 2011.

22 Alexander Sheh and Daniel P. W. Ellis. Chord segmentation and recognition using EM-trained hidden Markov models. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 185–191, Baltimore, USA, 2003.

23 Yushi Ueda, Yuuki 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 Acoustics, Speech and Signal Processing (ICASSP)*, pages 5518–5521, Dallas, USA, 2010.