# USING MUSICAL STRUCTURE TO ENHANCE AUTOMATIC CHORD TRANSCRIPTION

**Matthias Mauch, Katy Noland, Simon Dixon**
Queen Mary University of London, Centre for Digital Music
{matthias.mauch, katy.noland, simon.dixon}@elec.qmul.ac.uk

## ABSTRACT

Chord extraction from audio is a well-established music computing task, and many valid approaches have been presented in recent years that use different chord templates, smoothing techniques and musical context models. The present work shows that additional exploitation of the repetitive structure of songs can enhance chord extraction, by combining chroma information from multiple occurrences of the same segment type. To justify this claim we modify an existing chord labelling method, providing it with manual or automatic segment labels, and compare chord extraction results on a collection of 125 songs to baseline methods without segmentation information. Our method results in consistent and more readily readable chord labels and provides a statistically significant boost in label accuracy.

## 1. INTRODUCTION

The automatic extraction of chords from audio has applications in music retrieval, cognitive musicology, and automatic generation of lead sheets. In this work we present a technique that allows us to generate more authentic lead sheets than previously possible with automatic methods, by making use of musical structure. Much of musical structure is defined by repetition, a core principle in music [1, p. 229].

In popular songs a repeated *verse-chorus* format is common, in which the chord sequence is the same in all sections of the same type. In lead sheets, for better readability these sections would normally only be notated once, with repeats indicated. Our method mirrors this improvement by assigning the same chord progression to repeated sections. In addition, having found repeating sections, we have available several instances of a given chord sequence from which to estimate the chords, so we expect an improvement in estimation accuracy. We demonstrate the improvements in readability and accuracy using manually-annotated descriptions of the musical structure, and show that the improvement can also be achieved using an automatic structure annotation algorithm tailored to the task.

In Section 2 we describe related work. In Section 3 we describe the chord extraction method used and present a new segmentation technique that is tailored to our task of finding repeated chord sequences. We give examples of chord estimation with and without the segmentation technique in Section 4, and present quantitative chord estimation results in Section 5. In Section 6 we discuss our findings, and present our conclusions in Section 7.



## 2. RELATED WORK

The majority of approaches to automatic chord estimation rely on framewise chroma features [2] as a representation of the relative energy in each pitch class for a given time window, then apply some further processing to estimate the chords. When template-matching is used to identify chords, additional smoothing over time, for example by a median filter [3], is necessary due to musical variation and noise. Inference in hidden Markov models (HMMs) [4] simultaneously performs template-matching and smoothing. These methods treat chords as isolated features of the music, which is a considerable simplification. In reality, chords are heard in context, together with the melody, key, rhythm, form, instrumentation, and other attributes. Some chord estimation methods account for additional musical attributes during the estimation process such as key [5], or key and rhythm together [6, 7], which is a step towards a unified music analysis model.

In this work we extend the concept of unified music analysis by using repetition in the structure to enhance chord estimation. Dannenberg [8] shows that knowledge of the musical structure can greatly improve beat tracking performance, but to our knowledge the principle has not yet been applied to chord estimation.

Previous automatic music structure extraction techniques include those that primarily search for section boundaries, indicated by a sudden change in the feature of interest, which could be timbre [9], spectral evolution [10], or combinations of features [11]. A common approach is to cluster together frames that are similar, then label contiguous similar frames as a segment. However, this relies on a particular feature remaining approximately constant for the duration of a section. We are interested in chords, which do change during a section, so an approach that searches for repeated progressions [12, 13] is more appropriate for our purposes. Methods using this paradigm rely on a self-

similarity matrix [14], which is a symmetric, square matrix that contains a measure of the similarity between every pair of frames. Repeated sections appear as parallel diagonal lines, and can be extracted with some post-processing, such as application of a low pass filter to reduce noise [15] followed by a thresholding operation to find contiguous frames with high similarity. In Section 3.3 we present a new variation which is similar to algorithms proposed by Ong [16] and Rhodes and Casey [17] and extracts repeated chord progressions of equal length.

## 3. METHOD

In a song, we call a chord sequence that describes a section such as the verse or chorus a *segment type*. Any segment type may occur one or more times in a song and we call each occurrence a *segment instance*. To make use of segment repetition as part of the chord estimation process, we rely on segment types whose instances are not only harmonically very similar, but also have the same length in beats (see Section 3.4). This is not required of a general purpose segmentation task, and hence generic segmentations are not directly utilisable. In Section 3.2 we describe how we preprocess manual segmentations to meet our needs. For automatic segmentation we choose to implement our own algorithm, which fulfills the above requirements by design (Section 3.3). First, we describe the method for calculating our basic features, beat-synchronous chromagrams (Section 3.1).

### 3.1 Beat-Synchronous Chromagrams

The automatic segmentation and chord estimation algorithms both rely on chroma features that are synchronised to the musical beat. The features represent the importance of each pitch class at the current beat. The initial, short chroma frames are generated from a note salience representation similar to a constant-Q transform, at a hopsize of 512 samples (46 ms) from audio that has been downsampled to 11025 Hz. For the chord extraction algorithm we split the salience representation to obtain separate bass and treble chromagrams, but the chromagram used by the segmentation algorithm covers both the bass and the treble range. For details see [18].

In order to produce beat-synchronous chromagrams we obtain a single chroma vector for each beat by taking the median (in the time direction) over all the chroma frames falling between two consecutive beat times. We use one of two sorts of beat times: manual or automatic. The collection of manual beat annotations covers 125 songs performed by the rock group The Beatles. The automatic beat times were extracted using Davies's automatic beat-tracker [19] on the same set of songs.

### 3.2 Manual Structural Segmentation

The manual structural segmentations cover the same 125 songs by The Beatles as we have beat annotations for: 29 songs were annotated for a previous project [1], and 96 were newly annotated for the present work. The basis for all annotations are Pollack's song analyses [20].

Every song contains several segment types, some of which have multiple instances. In some songs, the instances of a segment type differ in length. In that case, to fulfill the requirement of equal length instances, the segment type is divided to create one or more new segment types whose instances all have the same length. This may result in new segment types having only one instance in the song.

### 3.3 Automatic Segmentation Algorithm

The automatic segmentation method has two main steps: finding approximately repeated chroma sequences in a song, and a greedy algorithm to decide which of the sequences are indeed segments. We calculate the Pearson correlation coefficients between every pair of chroma vectors, which together represent a beat-wise self-similarity matrix $R = (r_{ij})$ of the whole song. This is similar to the matrix of cosine distances used by Ong [16]. In the similarity matrix, parallel diagonal lines indicate repeated sections of a song. In order to eliminate short term noise or deviations we run a median filter of length 5 (typically just more than one bar) diagonally over the similarity matrix. This step ensures that *locally* some deviation is tolerated.

We perform a search of repetitions over all diagonals in the matrix over a range of lengths. We assume a minimum length of $m_1 = 12$ beats and a maximum length of $m_M = 128$ beats for a segment, leading to a very large search space. We minimise the number of elements we have to compare by considering as section beginnings only those beats that have a correlation $r$ greater than a threshold $t_r$, and assuming that section durations are quantised to multiples of four beats. We found that a value of $t_r = 0.65$ worked well. In future work we would like to learn $t_r$ from data. We further reduce the search space by allowing segments to start only at likely bar beginnings. Likely bar beginnings are beats where the convolution of a function representing the likelihood of a change in harmony, and a kernel with spikes every two beats has a local maximum (details in [18]).

To assess the similarity of a segment of length $l$ starting at beat $i$ to another one of the same length starting at $j$ we consider the diagonal elements

$$D_{i,j,l} = (r_{i,j}, r_{i+1,j+1}, \ldots, r_{i+l,j+l}) \qquad (1)$$

of the matrix $\mathcal{R}$. If the segments starting at $i$ and $j$ are exactly the same, then $D_{ij}$ will be a vector of ones, and hence we can characterise a perfect match by

$$\min\{D_{i,j,l}\} = 1. \qquad (2)$$

To accomodate variation arising in a practical situation, we relax the requirement (2) by using the empirical $p$-

[1] Segmentations available at `http://www.elec.qmul.ac.uk/digitalmusic/downloads/index.html#segment`.

quantile function [2] instead of the minimum (which is the 0-quantile), and choosing a segment threshold $t_s$ lower than unity. The triple $(i, j, l)$ hence describes a repetition, if

$$\text{quantile}_p\{D_{i,j,l}\} > t_s. \quad (3)$$

The two parameters $p = 0.1$ and $t_s = 0.6$ are chosen empirically. In future work we would like to learn these values from the ground truth data. The set of repetitions $\mathcal{R}_{il} = \{j : \text{quantile}_p\{D_{i,j,l}\} > t_s\}$ is then added to a list $\mathcal{L}$ of repetition sets, if it has more than one element $j$, i.e. if it actually describes at least one repetition. If two segments $(i, j_1, l)$ and $(i, j_2, l)$ overlap, only the index of the one with the higher score is retained in $\mathcal{R}_{il}$.

Each of the sets $\mathcal{R}_{il}$ represent a potential segment type, and its elements represent the start beats of instances of that segment type. However, there are typically many more repetition sets than there are segment types. To find repetition sets relating to actual segment types we use the heuristic of a music editor who tries to save paper: he will first take the repetition set in which $l \times |\mathcal{R}_{il}|$ is maximal, and then repeat this kind of choice on the remaining segments of the song, resulting in a greedy algorithm. The only exception to that rule is the case in which he finds that a sub-segment of a repetition is repeated more often than the whole segment. He then chooses the $\mathcal{R}_{il}$ pertaining to the sub-segment.

### 3.4 Using Repetition Cues in Chord Extraction

We use structural segmentation to combine several instances of a segment type in a song and then infer a single chord sequence from the combination.

The baseline is an existing chord labelling method [6], which extracts chords from beat-synchronous treble and bass chromagrams. Using a dynamic Bayesian network [21] similar to a hierarchical hidden Markov model the network jointly models metric position, chords and bass pitch class and infers the most probable sequence from the beat-synchronous chromagrams of the whole song. The method models four different chord classes: major, minor, diminished and dominant [3].

In order to integrate the knowledge of repeating segments, we split the chromagram for the whole song into smaller chromagram chunks, each belonging to one segment instance. If a segment type has more than one instance, all its chromagram chunks are averaged by taking the mean of the respective elements, thus creating a new chromagram chunk representing all instances of the segment type. The chord extraction is then performed on the newly generated chromagram chunk, and the estimated chords are transcribed as if they had been extracted at the individual segment instances.

## 4. EXAMPLES

In this section we present some example chord transcriptions with and without the segmentation technique, for the fully automatic method. Figure 1 shows a complete song segmentation, and indicates regions where the chord extraction was correct with and without the segmentation technique. Figures 2 and 3 show some excerpts on a larger scale, with the chord estimation detail visible. It is clear that the segmentation technique has had a defragmentation effect on the chord labels. A change in realisation of a repeated chord sequence between segment instances, such as a difference in melody, has in numerous places caused the standard transcription to incorrectly change chord, but when repeated segments are averaged these inconsistencies are removed. Examples include the `E:min` chord in the third row of Figure 2 and the fragmented `F♯` chords in the third row of Figure 3. This not only improves the chord accuracy (see Section 5), but also results in more natural transcriptions that include repeated chord progressions, so could be used to generate compact lead-sheets with each segment written exactly once. The figures demonstrate how the segmentation technique generates chord progressions that are indeed identical for all instances of a given segment type.

For a few songs the segmentation caused the chord estimation accuracy to decrease. Figure 4 shows an excerpt from *A Taste of Honey*, a song with one of the greatest reductions in chord accuracy due to segmentation. The transcription in the second row is good in general, but the long F sharp minor chord has been incorrectly labelled as major, an error that repeats three times in the song. The final chord in the song is F sharp major, and the segmentation algorithm has incorrectly marked this chord as a repetition of the minor chords earlier on. The problem is compounded by the behaviour of the automatic beat tracker at the end of the song: when the true beats stop, the beat tracker continues at a much faster tempo, which has caused the last chord to appear to have the same length in beats as the much longer (in seconds) F sharp minor chords throughout the song. This poor case, then, still produces a good transcription but with a parallel major-minor error caused in part by the beat tracker giving too much importance to the final chord.

## 5. QUANTITATIVE RESULTS

While the previous section has demonstrated how segmentation can help create consistent and more readily readable chord transcriptions, this section examines their overall performance. To that end we compare the six different combinations arising from two different beat annotations (manual and automatic) and three different segmentation annotations (manual, automatic, and none).

For each of the ground truth chords, we make a musical judgement regarding whether it should fall into one of the chord classes we investigate: major, minor, diminished, dominant or no chord. If there is no clear suitable mapping, for example for an augmented chord, our chord estimation will always be treated as incorrect. We use as an evaluation measure the relative correct overlap per song in physical time against a reference of Harte's chord tran-

[2] `http://www.mathworks.com/access/helpdesk/help/toolbox/stats/quantile.html`

[3] strictly speaking: major with a minor seventh

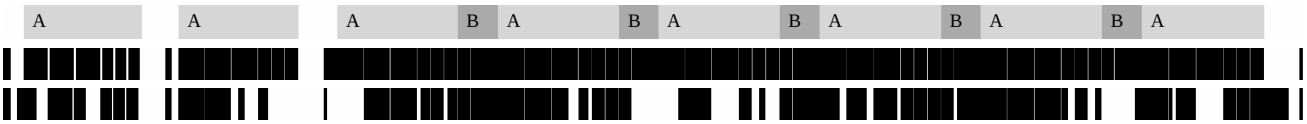


**Figure 1**. *Dizzy Miss Lizzy* (complete). First row: automatic segmentation. Second row: regions of correctly-labelled chords using segmentation. Third row: regions of correctly-labelled chords without using segmentation.

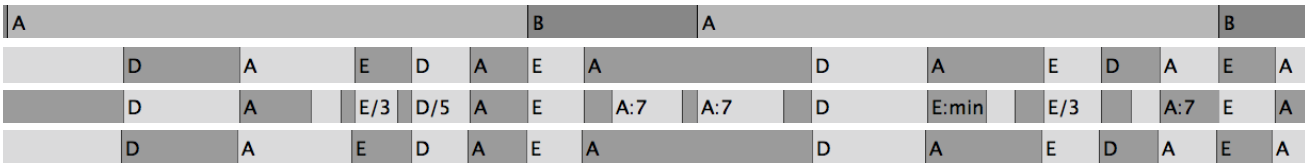


**Figure 2**. Extract from *Dizzy Miss Lizzy*. First row: automatic segmentation. Second row: automatic chord labels using segmentation. Third row: automatic chord labels without using segmentation. Fourth row: hand-annotated chord labels.

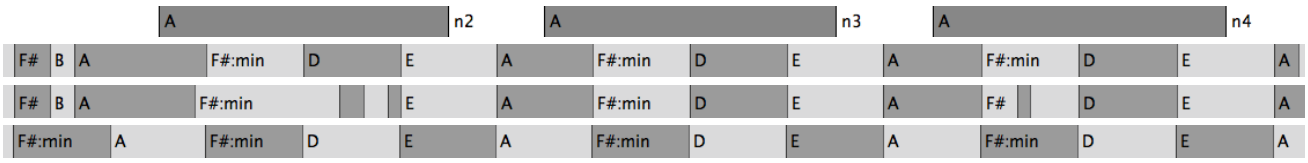


**Figure 3**. Extract from *Please Mister Postman*. First row: automatic segmentation. Second row: automatic chord labels using segmentation. Third row: automatic chord labels without using segmentation. Fourth row: hand-annotated chord labels.

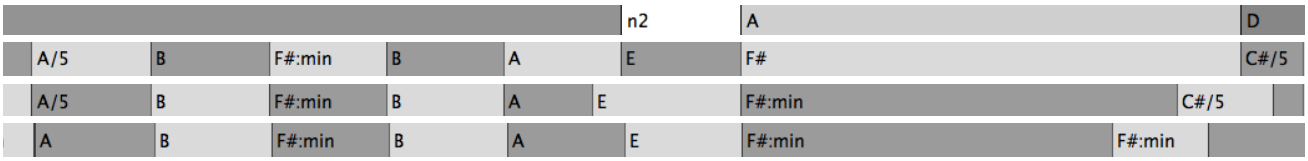


**Figure 4**. Extract from *A Taste of Honey*. First row: automatic segmentation. Second row: automatic chord labels using segmentation. Third row: automatic chord labels without using segmentation. Fourth row: hand-annotated chord labels.

scriptions [22], i.e.

$$\mathcal{O} = \frac{\text{summed duration of correct chords}}{\text{duration of song}}. \qquad (4)$$

A chord is considered correct if its chord type matches that of the ground truth chord and its root note matches that of the ground truth or its enharmonic equivalent. In Table 1 we report mean overlap scores over the 125 songs. For completeness we also report the equivalent scores using the chord classes used in the MIREX chord detection task [23], in which only two chord classes are distinguished. We recommend that these numbers are used only to assess the approximate performance of the algorithm because—as can be seen in Figure 5—the distribution is multimodal with a wide spread, due to the large range of difficulty between songs. An evaluation method that takes into account these "row effects" is the Friedman analysis of variance [24] based on ranking the results per song. The associated $p$-value is below double precision, suggesting that at least one method is significantly different from the others. The multiple comparison analysis [4] in Figure 6 shows that the improvements due to segmentation cues for both manual segmentation and automatic segmentation are significant. Figure 7 illuminates why this is so: the use of segmentation information leads to an improved relative overlap score in most of the songs, for example, automatic segmentation improves accuracy on 74% of songs.

Table 1 shows that the choice of segmentation method makes very little difference to our results, with a much greater difference caused by the beat annotation method. Since the automatic beat tracker was adjusted for quick tempos, several songs were tracked at double tempo with respect to the manual annotations, so our results suggest that the chord estimation method works better with higher beat granularity.

## 6. DISCUSSION

The method presented here is not tied to the individual algorithms. Using other chord extraction or segmentation methods could further improve results and shed more light on the performance of its constituent parts. As mentioned in Section 3.3 we plan to investigate the effects of training some of the segmentation parameters. It would also be in-

[4] `http://www.mathworks.com/access/helpdesk/help/toolbox/stats/multcompare.html`

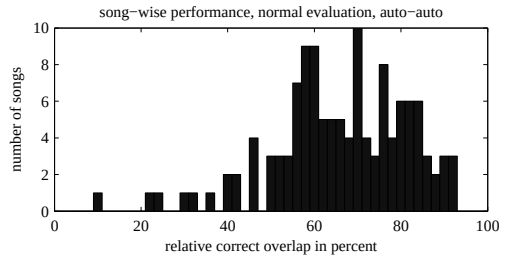


**Figure 5**. Relative correct overlap for the configuration using automatic beats and automatic segmentation: Histogram showing song frequencies. The clearly non-Gaussian distribution suggests that the mean correct overlap should not be the main evaluation technique.

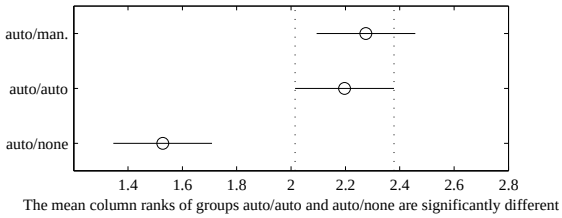


**Figure 6**. Multiple comparison test of the three best-performing variants (automatic beat extraction) at a confidence level of 99%, based on Friedman analysis of variance. The upper two rows show that of the two methods using manual (auto/man.) and automatic (auto/auto) segmentation significantly outperform the one without (auto/none), while the difference between automatic and manual segmentation is not significant.

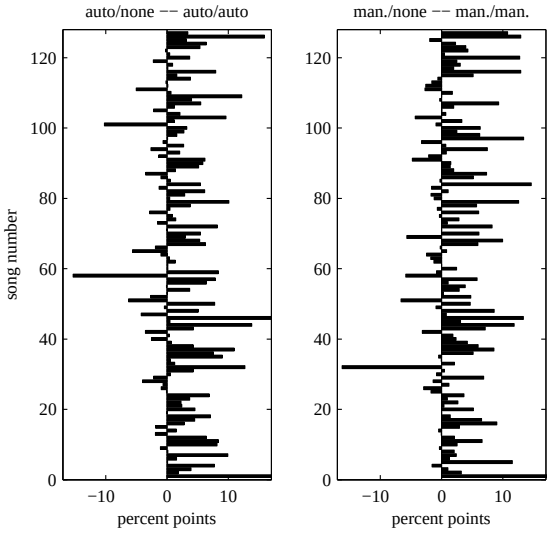


**Figure 7**. Song-wise improvement in correct relative overlap for the methods using segmentation cues: using automatic beats, automatic segmentation improves performance on 74% of songs (left); for manual beats, manual segmentation improves 68% of songs (right).

| configuration | | four classes | MIREX |
|---|---|---|---|
| man. beat | man. segm. | 64.4 | 71.8 |
| | auto segm. | 64.1 | 71.5 |
| | no segm. | 61.7 | 69.1 |
| auto beat | man. segm. | 66.4 | 73.7 |
| | auto segm. | 65.9 | 73.0 |
| | no segm. | 63.4 | 70.7 |

**Table 1**. Mean relative overlap in percent and mean rank results. The four classes measure is our preferred measure for this task. The MIREX measure gets higher scores, since it maps all chords to two classes, in particular dominant and major chords are taken to be equivalent.

teresting to determine whether using the median (instead of the mean) to average chromagram chunks would lead to improvements for cases like *A Taste of Honey*, where one major chord has tipped the mean to the parallel major.

The present work focussed on early rock music. We expect that—given a good segmentation—improvements in recognition results could be even greater for jazz: while the extraction of chords in jazz is more difficult than in rock music due to improvisation and more complex chord types, the repetition of segment types is often more rigid.

The method to share information globally between segments we used for this work is a simple one. Integrating this process with the chord extraction itself is a more elegant solution, but would require structure learning.

## 7. CONCLUSIONS

We have shown that using knowledge of repeating structure in a song can improve chord recognition in two ways. Firstly, by design the chord estimates are more consistent between instances of the same segment type, which leads to a more natural transcription that could be used to generate realistic lead sheets with structure markings. Secondly, we have shown that our method of averaging the different instances of each segment type has significantly improved the measured chord accuracy. This is demonstrated by examples that show how non-repeating incorrect chord fragments are removed by the averaging process. The improvement is observed both when using manually-annotated beat times and segments, which shows that the principle is valid, and when using a fully-automatic method, which shows that the principle can be applied to real systems, and is effective even when there are some errors in the beat or segment labels.

The results we have presented support the wider hypothesis that unified music analysis improves estimation of individual features [6–8]. We would like to extend this approach in our future work to allow chord estimation to be informed by a complete musical context, including melody, tonality, timbre and metrical structure.

## 8. REFERENCES


[1] David Huron. *Sweet Anticipation: Music and the Psychology of Expectation*. MIT Press, 2006.

[2] Takuya Fujishima. Real time chord recognition of musical sound: a system using Common Lisp Music. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 464–467, 1999.

[3] Christopher Harte and Mark Sandler. Automatic chord identifcation using a quantised chromagram. In *Proceedings of 118th Convention*. Audio Engineering Society, 2005.

[4] Juan P. Bello and Jeremy Pickens. A Robust Mid-level Representation for Harmonic Content in Music Signals. In *Proceedings of the 6th International Conference on Music Information Retrieval, ISMIR 2005, London, UK*, pages 304–311, 2005.

[5] Kyogu Lee and Malcolm Slaney. Acoustic Chord Transcription and Key Extraction From Audio Using Key-Dependent HMMs Trained on Synthesized Audio. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2):291–301, February 2008.

[6] Matthias Mauch and Simon Dixon. Simultaneous estimation of chords and musical context from audio. To be published in IEEE Transactions on Audio, Speech, and Language Processing.

[7] Hélène Papadopoulos and Geoffroy Peeters. Simultaneous estimation of chord progression and downbeats from an audio file. In *Proceedings of the 2008 ICASSP Conference*, pages 121–124, 2008.

[8] Roger B. Dannenberg. Toward automated holistic beat tracking, music analysis, and understanding. In *Proceedings of the 6th International Conference on Music Information Retrieval*, London, 2005.

[9] S. Abdallah, K. Noland, M. Sandler, M. Casey, and C. Rhodes. Theory and evaluation of a Bayesian music structure extractor. In *Proceedings of the 6th International Conference on Music Information Retrieval, ISMIR 2005, London, UK*, pages 420–425, 2005.

[10] G. Peeters, A. La Burthe, and Xavier Rodet. Toward automatic music audio summary generation from signal analysis. In *Proceedings of the 3rd International Conference on Music Information Retrieval*, Paris, 2002.

[11] Namunu C. Maddage. Automatic structure detection for popular music. *IEEE Multimedia*, 13(1):65–77, 2006.

[12] Meinard Müller and Frank Kurth. Towards structural analysis of audio recordings in the presence of musical variations. *EURASIP Journal on Advances in Signal Processing*, 2007.

[13] Masataka Goto. A chorus-section detecting method for musical audio signals. In *Proceedings of the 2003 IEEE Conference on Acoustics, Speech and Signal Processing*, pages 437–440, 2003.

[14] Jonathan Foote. Visualizing music and audio using self-similarity. In *Proceedings of the 7th ACM International Conference on Multimedia (Part 1)*, pages 77–80, 1999.

[15] Mark A. Bartsch and Gregory H. Wakefield. Audio thumbnailing of popular music using chroma-based representations. *IEEE Transactions on Multimedia*, 7(4), February 2005.

[16] Bee Suan Ong. *Structural Analysis and Segmentation of Music Signals*. PhD thesis, Universitat Pompeu Fabra, 2006.

[17] Christophe Rhodes and Michael Casey. Algorithms for determining and labelling approximate hierarchical self-similarity. In *Proceedings of the 2007 ISMIR Conference, Vienna, Austria*, pages 41–46, 2007.

[18] Matthias Mauch. A chroma extraction method and a harmonic change detection function. Technical report, Queen Mary, University of London. Available at `http://www.elec.qmul.ac.uk/digitalmusic/papers/2009/Mauch09-C4DM-TR-09-05.pdf`.

[19] Matthew Davies. *Towards Automatic Rhythmic Accompaniment*. PhD thesis, Queen Mary University of London, London, UK, August 2007.

[20] Alan W. Pollack. Notes on... series, 1995. Available at `http://www.recmusicbeatles.com`.

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

[22] Christopher Harte, Mark Sandler, Samer A. Abdallah, and Emilia Gomez. Symbolic representation of musical chords: A proposed syntax for text annotations. In *Proceedings of the 6th International Conference on Music Information Retrieval, ISMIR 2005, London, UK*, pages 66–71, 2005.

[23] MIREX audio chord detection subtask, music information retrieval evaluation exchange, 2008. `http://www.music-ir.org/mirex/2008/index.php/Audio_Chord_Detection`.

[24] David Hull. Using statistical testing in the evaluation of retrieval experiments. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 329–338. ACM New York, USA, 1993.