Obtaining General Chord Types from Chroma Vectors

Marcelo Queiroz
Computer Science
Department
University of São Paulo
mqz@ime.usp.br

Maximos
Kaliakatsos-Papakostas
Department of Music Studies
Aristotle University of
Thessaloniki
maxk@mus.auth.gr

Emilios Cambouropoulos
Department of Music Studies
Aristotle University of
Thessaloniki
emilios@mus.auth.gr

ABSTRACT

This paper presents two novel strategies for processing chroma vectors corresponding to polyphonic audio, and producing a symbolic representation known as GCT (General Chord Type). This corresponds to a fundamental step in the conversion of general polyphonic audio files to this symbolic representation, which is required for enlarging the current corpus of harmonic idioms used for conceptual blending in the context of the COINVENT project. Preliminary results show that the strategies proposed produce correct results, even though harmonic ambiguities (e.g. between a major chord with added major 6th and a minor chord with minor 7th) might be resolved differently according to each strategy.

Categories and Subject Descriptors

[**Multimedia and Multimodal Retrieval**]: Music Retrieval

1. INTRODUCTION

This paper deals with processing chroma feature vectors obtained from audio databases containing polyphonic music, in order to automatically obtain General Chord Type (GCT) representations for music pieces [2]; this is a special case of chord transcription where GCTs are used instead of the more common guitar-style (e.g. Am, D7) chord annotations. GCT is a symbolic chord representation that decomposes chords into three fundamental parts: a root, a base (maximally consonant pitch-class subset) and extensions (additional pitch-classes). This representation combines the flexibility of pc-set representations for dealing with non-tonal music idioms with the expressiveness of traditional tonal harmony through a generic consonance representation framework. The problem of obtaining GCT representations from audio also relates to, but is considerably easier than,

polyphonic audio transcription, in the sense that the output does not require any timbre or voice-related information, that octave-equivalence may be explored for improved computational performance, and also due to the fact that GCTs offer some degree of robustness regarding transcription errors (e.g. doubled notes or wrong octaves do not affect the result).

The main interest in developing these methods lies in obtaining symbolic databases of music from different idioms that will be used in the contexts of automatic harmonization [9] and conceptual blending [22]. GCTs offer a more detailed account of the harmonic progressions than plain pitch-class sets, and this additional information (e.g. roots and dissonant notes) can be explored in the logical ontologies that are used for conceptual blending. The original algorithm for obtaining GCTs [2] depends on pre-existing symbolic representations of the chords (e.g. MIDI or pc-sets), and so GCT-based automatic harmonization and conceptual blending of music idioms depend on large databases of symbolic data for training and validation. One possibility of extending the repertoire available for tests and future research in these fields consists in obtaining GCTs directly from audio recordings.

We offer a preliminary account of a few potential methods for obtaining GCTs from audio based on the use of chroma feature vectors. Chroma features are computed for (usually very short) windowed audio frames in the context of the Short-Time Fourier Transform, and they represent in a 12-dimensional vector every pitch-class in the 12 equal-tempered octave division. Each chroma value accumulates spectral energy information from all octaves in the audible range corresponding to a particular pitch-class. These values are then used for extracting the relevant components of the GCT corresponding to a given frame.

1.1 Related Work

Automated chord transcription from audio is the task to identify chord labels on segments of musical audio signal. A first approach in chord transcription was presented in [6], where notes where extracted from audio and through the symbolic content described by notes, chord labels were assigned. The notes-to-chords matching was based on simple pattern matching based on chord templates, which are vectors that describe the intensity of each pitch class. Chord templates have also been used for chord transcription in more recent works [19, 18], proposing more robust techniques for identifying chord-change locations and labelling the segmented parts.

Among the most successful techniques for chord transcrip-

tion is the additional utilisation of hidden Markov models (HMMs). The pioneering work in audio cord transcription with HMMs was presented in [23], where HMMs were employed on a dataset of 20 Beatles songs. To train and test the system, ground-truth labels of these pieces were collected from a book of Beatles transcriptions. Several other methods were proposed that have been characterised as "variations of the standard methods" [4] (meaning as standard methods the ones presented in [6] and [23]), some of which utilised HMMs with additional variations [13, 14, 12, 3], geometric characteristics of the chroma space [8] and recurrent neural networks [1].

This paper focuses on the quality of information that is embodied in audio signals in terms of isolated chord labels, skipping the task of segmenting audio and "filtering" the spectrogram (as a means to obtain clear pitch-related information). Even though the problem of interdependence between segmentation and labels [25] is an important aspect of effective chord transcription, aim of the paper at hand is to provide musicologically grounded insights about fundamental chord concepts, like consonance (examined through the General Chord Type (GCT) [2] representation) and perceived pitch hierarchies (expressed through Parncutt's theory [21]). To this end, the labelling quality of single-chord recordings is examined, by utilising spectral information from the chroma profile, extracted with the MATLAB Chroma Toolbox [15, 16, 17] and theoretic models of pitch-combination consonances expressed by the GCT model.

GCTs will be presented in some detail in Section 2.1, but for the sake of comparison with related work on obtaining guitar-style chord annotations, two aspects of GCTs have to be taken into account: structure and idiom-independence. GCTs are structured in the sense that they rely on a consonance model that allows it to recognize different parts of a chord, namely root, base (maximal consonant part) and dissonant extensions. Furthermore, these components, which of course exist in traditional harmonic notations, are not defined exclusively for western tonal harmony, but are inferred from the consonance model, which is flexible enough to be applied in widely different music idioms, such as Jazz, modal Folkloric music, western tonal Classical and Popular music and also 20th- and 21st-century Atonal music.

2. CONCEPTS AND METHODOLOGY

2.1 General Chord Types

The General Chord Type (GCT) representation [2] incorporates the concepts of root, base and extension of a chord. These concepts derive from traditional harmonic analysis and establish a hierarchy between the notes of a chord, where the root represents the fundamental note upon which the rest of the chord is built (in tonal music this construction would be based on stacking thirds, although the GCT does not presuppose such method), the base represent a subset of pitch-classes that are pairwise consonant according to a user-defined binary consonance vector[1] and which is maximal with respect to this property, and extension represents

---

[1] Consonance and dissonance in music are intricate concepts which are outside of the scope of this text. In contrast with the psychoacoustic notion of dissonance which includes timbre as an essential defining characteristic, the consonance vector here considered is arbitrary and reflects cultural aspects related to a given music idiom.

other notes aggregated to this collection which might be consonant or dissonant with respect to notes belonging to the base (but are necessarily dissonant with respect to at least one note in the base, otherwise they would belong to the base by definition).

Such a hierarchy is built upon an abstract model that takes as input a given scale and a partition of the interval space into consonant and dissonant intervals. The GCT representation obtained depends fundamentally on those idiom-specifuc definitions; for instance, by defining every possible interval in a chromatic (12-semitone-equal-tempered) scale, GCTs are equivalent to the normal orders of pc-sets [5], frequently used in the analysis of atonal music. On the other hand, by considering as consonant only unisons, thirds, sixths and perfect fourths and fifths, the GCTs obtained are very close to traditional harmonic notation based on roman numerals, which are considerably more expressive than plain pc-sets in the analysis of tonal repertoire (in this aspect the GCT is comparable in expressiveness and ease-of-use to the model of Harte and co-authors [7], although the latter is not adequate for representing atonal music). These two examples illustrate the versatility of the GCT model in dealing with music idiomatic differences: besides atonal and western tonal music, this model has been also used to represent Rock and Jazz pieces, and polyphonic music from Epirus [11]. This versatility is particularly welcome in the context of conceptual blending across music idioms [10], since the same representation may be used for widely differing harmonic material, and at the same time it is expressive with respect to singularities of each idiom, allowing for a much more consistent and comprehensive exploration of hybrid music idioms.

2.2 Chroma feature vectors for transcribing GCTs

By definition, GCTs are computed from a symbolic representation of the chords [2], and in this sense the problem of obtaining them directly from audio input could be decomposed naturally into two steps: polyphonic audio transcription, which obtains symbolic music representations from audio, and GCT computation from these. Regrettably, there are good reasons against adopting such a strategy: computationally, polyphonic transcription is a very intensive task, which by state-of-the-art methods still provides inaccurate results, being often dependent on pre-existing timbre models or instrument spectral templates (although some techniques, such as non-negative matrix factorization, aim at finding those templates simultaneously to actual transcription [24]). The use of chroma vectors provides a reasonable alternative to GCT computation which bypasses polyphonic transcription and allows the direct identification of GCT components.

Chroma vectors are audio features that describe the harmonic content of an audio signal in terms of intensity or amplitude of spectral components corresponding to pitch-classes. Typically they are computed for short audio frames and for the 12-semitone equal-tempered scale, by adding spectral peaks corresponding to each pitch-class over all octaves in the audible range. The Chroma Toolbox is a Matlab implementation that provides basic signal processing functions as well as graphical output in the form of chroma-grams (the equivalent of spectrograms or sonograms but using chroma vectors as the vertical axis) [17].

Experimental validation of the techniques here presented is done by using as ground truth an annotated chord dataset which is synthesized in order to provide an automatically annotated audio database. By adopting such a strategy we can obtain GCTs directly from the original symbolic database, which are compared to the ones obtained from the synthesized audio input through chroma feature vectors. This strategy will also allow the use of information retrieval techniques such as cross-validation (by using part of the database as training set and part as experimental input data) and analysis of the results through measures such as precision and recall in future experiments.

2.3 Identifying the GCT root and base notes

The problem of identifying the main part of the chord, comprising root and base notes, is considered from the perspective of the chroma vector, which is directly obtained from a Fourier transform of a windowed portion of the signal. In this sense, chroma vectors are affected by the position of the notes within the audible spectrum, and this relates to orchestration: notes produced by spectrally richer instruments, as well as notes produced in the lowest register, will add more energy to the chroma vector, and this energy will appear not only in the index corresponding to the pitch-class, but also in its entire harmonic series. For this reason it is not always the case that the root will have more energy than other notes, as it is also not the case that the bass note will have necessarily more energy. Algorithmic strategies have to be flexible regarding how to select relevant pitch-classes from the chroma vector.

In order to select a reasonable subset of pitch-classes from which to search for the root and base of the chord, the consonance vector can be used to provide a threshold of chroma energy. Specifically, given a consonance vector it is easy to compute the maximum number N of different pitch-classes that can belong to any base of any chord; for instance, the tonal consonance vector $(1,0,0,1,1,1,0,1,1,1,0,0)$ allows no more than $N=3$ pairwise consonance pitches in any given chord. Because of the nature of the chroma vector it would be risky to assume that exactly the N pitches with highest chroma values would correspond to the base of the chord; instead we propose a strategy for selecting a larger set of pitch classes. This may be viewed as an intermediate approximate transcription step which aims at obtaining a superset containing the root and base notes, where extra notes will be discarded by the GCT algorithm. This pre-processing of the chroma vector is given by algorithm 1 below.

```
Algorithm 1: Approximate Transcription
input: chroma vector c, indexed from 0 to 11.
output: a set of pitch classes
step 1: sort c in ascending order
step 2: compute the first derivative
                    d(i)=c(i)-c(i-1)
step 3: normalize d
step 4: select k = min { i | d(i)>T }
step 5: define N = 12-k
step 6: return the original indices of the
                    N largest chroma values
```

This algorithm depends on a parameter T which is a threshold on the normalized derivative values for identifying a cutting point in the sorted chroma vector and producing a set of pitch classes that desirably contains all the notes in the chord (and maybe others). Given this set, the original GCT algorithm [2] is used to sort out the pitch classes into root, base and extension.

We investigate two implementations of the GCT algorithm, the one originally proposed for dealing with purely symbolic data (where chroma values would be meaningless), and another one that incorporates information from chroma values. Specifically, there is a step in the GCT algorithm where it is confronted with competing hypothesis for root and base (e.g. the chord A-C-E-G can be viewed as C major with added 6th or as A minor with added 7th). Whereas the original algorithm has a rule based on the overlapping of base candidates, one variant here considered (referred to as Audio GCT) weighs chroma values according to Parncutt model for root-finding [20] and chooses the candidate with the highest score.

Since the transcription step is not perfect, but approximate, it makes no sense to keep the notes identified by the GCT algorithm as added dissonances. Instead we rely on the chroma values again for identifying the correct extension of the GCT.

2.4 Identifying the GCT Extension

After the identification of chord root and base notes, the problem reduces to selecting candidates among the remaining notes that are actually present in the chord being analyzed. This is not a completely trivial task for the fact that notes present in the input audio add nonzero energy to their entire harmonic series, so some nonzero chroma values might reflect pure harmonic energy but absent notes (e.g. the note B on a C major chord, which receives contributions from the harmonic series of both E and G).

If we assume that absent notes will have smaller energy values than dissonances actually present in the chord, then this problem becomes one of finding a suitable threshold, not unlike the approximate transcription step above. The main difference is that this step may benefit from information obtained in the identification of the base. Algorithm 2 presents this step using both the chroma vector and the base identified by the GCT algorithm in the previous step.

```
Algorithm 2: Finding GCT Extensions
input: chroma vector c, indexed from 0 to 11
                            and GCT base
output: a set of pitch classes corresponding
                            to GCT extensions
step 1: re-scale c by dividing it by the
                    mean value of base notes
step 2: remove base notes from c
step 3: sort c in ascending order
step 4: compute the first derivative
                        d(i)=c(i)-c(i-1)
step 5: return all indices i such that d(i)>T
```

This algorithm also depends on a threshold T on normalized derivative values, but the normalization here reflects the base identification step.

3. EXPERIMENTS AND DISCUSSION

The above strategies were tested on a set of chords that occur commonly in western tonal music: major and minor triads in open position, their inversions, and these chords with added major 6ths and with added minor 7ths. Since

all strategies are invariant by transposition, a common root (C) was chosen for all chords, resulting in twenty chord descriptions. These descriptions were synthesized using SOX and the Karplus-Strong plucked string algorithm available through SOX synth module (using 4 seconds of audio for each chord). Chroma vectors were obtained using the ChromaToolbox [15] implemented in MatLab, using the default parameters (0.1 seconds window with 50% overlap), and adding all chroma obtained over the 4 seconds of each file.

Two preliminary experiments were run for Algorithms 1 (approximate transcription) and 2 (identifying extensions), both using T=0.5 as threshold. Algorithm 1 alone was able to correctly transcribe the notes in the chord in 75% of the input chords, and in the remaining 25% one extra note was incorrectly transcribed. In no case the produced pc sets fail to contain the desired base nor the expected extensions of the chords. Algorithm 2 alone was tested using the manually annotated bases, and it correctly identified the desired extensions from the chroma vector in 100% of the cases. Evidently when these procedures are pipelined there is always the possibility of error propagation, so this 100% is only a confirmation of the adequacy of this isolated step.

Table 1 exhibits that both the original and the audio GCT variations identify the roots, bases and extension of the examined chords with high accuracy (over 80%) - although this percentage is indicative as later explained. This accuracy is dependent both on the part of the presented algorithm that decided about the PCs that are present in an audio segment as well as the part where the actual chord extensions are identified - while chroma values that belong to accumulations of harmonics of actual notes are discarded.

The GCT algorithm, being a methodology that relies on pitch classes, is expected to misinterpret "ambiguous" chord schemes, like for instance the pc-set $\{0, 3, 7, 10\}$. Since no further information is provided, this pc-set could either refer to a minor seventh chord with root 0 ($[0, [0, 3, 7], [10]]$), or a major sixth chord with root 3 ($[3, [0, 4, 7], [9]]$). On a theoretic basis, none of these choices is "incorrect". The original GCT algorithm systematically misinterprets the C6 chords as A♭m7, while in inversion C6 (G bass) the algorithm fails to identify the extension (due to the high value that corresponds to the PC 11 in the chroma vector, which smoothens the derivative towards the PC 10 peak).

In contrast, the audio GCT variation that utilises Parncutt's harmonic information to untie competing bases, misinterprets the minor seventh chords, exhibiting a preference towards major chord bases. Therefore, all four inversions of Cm7 are interpreted as E♭m7 chords, while their extensions, in this root and base context, are correctly identified. As previously mentioned, the accuracy percentages displayed in Table 1 are indicative in a sense that both GCT variations are actually performing *equally* well in identifying roots and bases in the examined examples. The difference in the accuracy results lies on the fact that three inversions were considered for the C6 chords (omitting the inversion that would put A in the bass), while four inversions were considered for the Cm7 chords. Therefore, the original GCT is shown to misinterpret three cases, while audio GCT four.

It is important to note that, except for the missing extension on chord C6 (G bass) which is an incorrect result by any standard, all results that were marked as incorrect corresponds to valid interpretations of the chords in terms of traditional harmonic analysis. In fact the two chord categories "major chords with added major 6th" and "minor chords with added minor 7th" are equivalent from the point of view of pc-set theory, being one the rotation of the other in the modulus-12 pitch representation space. Also from the perceptual point of view the audio examples are ambiguous, and each one can be heard one way or another depending on which note the listener focus his/her attention. The correct analysis of these chords in a given harmonic context would depend on previous and following chords, and also on voice leading information (e.g. melodic resolutions).

4. CONCLUSIONS

This paper presented two novel strategies for obtaining GCT representations from chroma vectors obtained from polyphonic audio input. This is a necessary step towards the automatic conversion of audio databases for polyphonic music into symbolic databases adequate for automatic harmonisation and conceptual blending.

The experimental section provide preliminary results that confirm the adequacy of the proposed strategies, yet further work is required on more general types of audio input. Particularly, it is important to consider different timbres and also different chord dispositions in terms of the frequency range.

Moving on to the problem of processing polyphonic audio input, a few problems are already identified and are the subject of future work:

- When processing a large audio file there are two choices to be explored: segment the audio into steady chords (e.g. by thresholding the derivative of consecutive chroma vectors) and then process each steady chord by one of the techniques presented herein, or use a fixed segmentation and produce a long chain of GCTs for each chord in the input, and then process this sequence to chunk identical or very similar GCTs into a single chord event. The first alternative is subjected to prior segmentation errors, which would produce wrong or cluttered GCTs, whereas the second alternative poses the problem of how to resolve ambiguities and differences of interpretation of GCTs obtained from different chroma vectors but belonging to the same chord (e.g. the C major with added major 6th and the A minor with added minor 7th).

- Symbolic databases for automatic harmonisation and conceptual blending are supposed to have a high-level analytical representation of the chords, aiming at representing the harmonic structure underlying the actual notes present in the score. Some melodic or counterpunctual devices should be ignored (e.g. passing notes) whereas others have a harmonic function (e.g. suspended and then resolved dissonances). This is a very difficult challenge in producing these symbolic databases, and current solutions depend on expert knowledge and manual annotations. Some form of semi-automatic pre-processing of these situations, trying to identify and label at least the most recurring forms of these devices, would be of great help in producing those databases.

chord	extracted PCs	GCT on pitch classes			GCT with audio information		
		root	base	extensions	root	base	extensions
C (C bass)	$\{0,4,7\}$	0	[0 4 7]	[ ]	0	[0 4 7]	[ ]
C (E bass)	$\{0,4,7\}$	0	[0 4 7]	[ ]	0	[0 4 7]	[ ]
C (G bass)	$\{0,4,7\}$	0	[0 4 7]	[ ]	0	[0 4 7]	[ ]
C7 (C bass)	$\{0,4,7,10\}$	0	[0 4 7]	[10]	0	[0 4 7]	[10]
C7 (E bass)	$\{0,4,7,10\}$	0	[0 4 7]	[10]	0	[0 4 7]	[10]
C7 (G bass)	$\{0,4,7,10,11\}$	0	[0 4 7]	[10]	0	[0 4 7]	[10]
C7 (B♭ bass)	$\{0,4,7,10\}$	0	[0 4 7]	[10]	0	[0 4 7]	[10]
C6 (C bass)	$\{0,4,7,9\}$	9	[0 3 7]	[10]	0	[0 4 7]	[9]
C6 (E bass)	$\{0,4,7,9\}$	9	[0 3 7]	[10]	0	[0 4 7]	[9]
C6 (G bass)	$\{0,4,7,9,11\}$	9	[0 3 7]	[ ]	0	[0 4 7]	[9]
Cm (C bass)	$\{0,3,7\}$	0	[0 3 7]	[ ]	0	[0 3 7]	[ ]
Cm (E♭ bass)	$\{0,3,7\}$	0	[0 3 7]	[ ]	0	[0 3 7]	[ ]
Cm (G bass)	$\{0,3,7\}$	0	[0 3 7]	[ ]	0	[0 3 7]	[ ]
Cm6 (C bass)	$\{0,3,7,9\}$	0	[0 3 7]	[9]	0	[0 3 7]	[9]
Cm6 (E♭ bass)	$\{0,3,7,9\}$	0	[0 3 7]	[9]	0	[0 3 7]	[9]
Cm6 (G bass)	$\{0,3,7,9,11\}$	0	[0 3 7]	[9]	0	[0 3 7]	[9]
Cm7 (C bass)	$\{0,3,7,10\}$	0	[0 3 7]	[10]	3	[0 4 7]	[9]
Cm7 (E♭ bass)	$\{0,3,7,10\}$	0	[0 3 7]	[10]	3	[0 4 7]	[9]
Cm7 (G bass)	$\{0,3,7,10,11\}$	0	[0 3 7]	[10]	3	[0 4 7]	[9]
Cm7 (B♭ bass)	$\{0,3,7,10\}$	0	[0 3 7]	[10]	3	[0 4 7]	[9]
		accuracy: 85%			accuracy: 80%		

Table 1: Chord labels assigned by the examined approaches. Misinterpretations appear in grey background.

5. REFERENCES

[1] N. Boulanger-Lewandowski, Y. Bengio, and P. Vincent. Audio chord recognition with recurrent neural networks. In *Proceedings of the 13th International Society for Music Information Retrieval Conference, (ISMIR 2012)*, pages 335–340, Porto, Portugal, October 8-12 2013.

[2] E. Cambouropoulos, M. Kaliakatsos-Papakostas, and C. Tsougras. An idiom-independent representation of chords for computational music analysis and generation. In *Proceeding of the joint 11th Sound and Music Computing Conference (SMC) and 40th International Computer Music Conference (ICMC)*, ICMC–SMC 2014, 2014.

[3] R. Chen, W. Shen, A. Srinivasamurthy, and P. Chordia. Chord recognition using duration-explicit hidden markov models. In *Proceedings of the 13th International Society for Music Information Retrieval Conference, (ISMIR 2012)*, pages 445–450, Porto, Portugal, October 8-12 2012.

[4] T. Cho, R. J. Weiss, and J. P. 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, 2010.

[5] A. Forte. *The structure of atonal music*. Yale University Press, New Haven, 1973.

[6] T. Fujishima. Realtime of musical sound : a system using common lisp music. In *Proceedings of the International Computer Music Conference, (ICMC 1999)*, pages 464–467, Bejing, China, October 22–27, 1999.

[7] C. Harte, M. Sandler, S. A. Abdallah, and E. Gómez. Symbolic representation of musical chords: A proposed syntax for text annotations. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR)*, pages 66–71, London, UK, 2005.

[8] C. Harte, M. Sandler, and M. Gasser. Detecting harmonic change in musical audio. In *Proceedings of the 1st ACM workshop on Audio and music computing multimedia*, AMCMM '06, pages 21–26, New York, NY, USA, 2006. ACM.

[9] M. Kaliakatsos-Papakostas and E. Cambouropoulos. Probabilistic harmonisation with fixed intermediate chord constraints. In *Proceeding of the joint 11th Sound and Music Computing Conference (SMC) and 40th International Computer Music Conference (ICMC)*, ICMC–SMC 2014, 2014.

[10] M. Kaliakatsos-Papakostas, E. Cambouropoulos, K.-U. Kühnberger, O. Kutz, and A. Smaill. Concept Invention and Music: Creating Novel Harmonies via Conceptual Blending. In *In Proceedings of the 9th Conference on Interdisciplinary Musicology (CIM2014)*, CIM2014, December 2014.

[11] M. Kaliakatsos-Papakostas, A. Katsiavalos, C. Tsougras, and E. Cambouropoulos. Harmony in the polyphonic songs of epirus: Representation, statistical analysis and generation. In *4th International Workshop on Folk Music Analysis (FMA) 2014*, June 2014.

[12] M. Khadkevich and M. Omologo. Use of hidden markov models and factored language models for

automatic chord recognition. In *Proceedings of the 10th International Society for Music Information Retrieval Conference, (ISMIR 2009)*, pages 561–566, Kobe, Japan, October 26-30 2009.

[13] K. Lee and M. Slaney. Automatic chord recognition from audio using a hmm with supervised learning. In R. Dannenberg, K. Lemstrom, and A. Tindale, editors, *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006)*, pages 133–137. University of Victoria, Oct. 2006.

[14] K. Lee and M. Slaney. A Unified System for Chord Transcription and Key Extraction Using Hidden Markov Models. In S. Dixon, D. Bainbridge, and R. Typke, editors, *8th International Conference on Music Information Retrieval ISMIR'07*, pages 245–250, Vienna, Austria, Sept. 2007. Österreichische Computer Gesellschaft.

[15] Meinard MÂ̈yller. Chroma Toolbox: Pitch, Chroma, CENS, CRP. http://www.mpi-inf.mpg.de/ mmueller/chromatoolbox/, 2010.

[16] M. Müller and S. Ewert. Towards Timbre-Invariant audio features for Harmony-Based music. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3):649–662, Mar. 2010.

[17] M. Müller and S. Ewert. Chroma Toolbox: MATLAB implementations for extracting variants of chroma-based audio features. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR)*, pages 215–220, Miami, USA, 2011.

[18] L. Oudre, C. Févotte, and Y. Grenier. Probabilistic framework for template-based chord recognition. In *Multimedia Signal Processing (MMSP), 2010 IEEE International*, pages 183 –187, Saint-Malo, France, Oct. 2010.

[19] L. Oudre, Y. Grenier, and C. Févotte. Template-based chord recognition: Influence of the chord types. In *Proceedings of the 10th International Society for Music Information Retrieval Conference, (ISMIR 2009)*, pages 153–158, Kobe, Japan, October 26-30 2009.

[20] R. Parncutt. *Harmony: A Psychoacoustical Approach.* Springer Berlin Heidelberg, Berlin, Heidelberg, 1989.

[21] R. Parncutt. A model of the perceptual root(s) of a chord accounting for voicing and prevailing tonality. In M. Leman, editor, *Music, Gestalt, and Computing*, volume 1317 of *Lecture Notes in Computer Science*, pages 181–199. Springer Berlin Heidelberg, 1997.

[22] M. Schorlemmer, A. Smaill, K.-U. Kühnberger, O. Kutz, S. Colton, E. Cambouropoulos, and A. Pease. Coinvent: Towards a computational concept invention theory. In *5th International Conference on Computational Creativity (ICCC) 2014*, June 2014.

[23] A. Sheh and D. P. Ellis. Chord segmentation and recognition using em-trained hidden markov models. In *International Symposium on Music Information Retrieval*, pages 185–191. ISMIR 2003, 2003.

[24] P. Smaragdis and J. C. Brown. Non-negative matrix factorization for polyphonic music transcription. In *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.*, pages 177–180. IEEE, 2003.

[25] T. Yoshioka, T. Kitahara, K. Komatani, T. Ogata, and H. G. Okuno. Automatic chord transcription with concurrent recognition of chord symbols and boundaries. In *Proceedings of the 5th International Society for Music Information Retrieval Conference, (ISMIR 2004)*, Barcelona, Spain, October 10-14, 2004.