CONCURRENT ESTIMATION OF CHORDS AND KEYS FROM AUDIO

Thomas Rocher, Matthias Robine, Pierre Hanna
LaBRI, University of Bordeaux
351 cours de la Libration
33405 Talence Cedex, France
{rocher,robine,hanna}@labri.fr

Darrell Conklin
Universidad del País Vasco
San Sebastián, Spain
and IKERBASQUE
Basque Foundation for Science
conklin@ikerbasque.org

ABSTRACT

This paper proposes a new method for local key and chord estimation from audio signals. A harmonic content of the musical piece is first extracted by computing a set of chroma vectors. Correlation with fixed chord and key templates then selects a set of key/chord pairs for every frame. A weighted acyclic harmonic graph is then built with these pairs as vertices, and the use of a musical distance to weigh its edges. Finally, the output sequences of chords and keys are obtained by finding the best path in the graph.

The proposed system allows a mutual and beneficial chord and key estimation. It is evaluated on a corpus composed of Beatles songs for both the local key estimation and chord recognition tasks. Results show that it performs better than state-of-the art chord analysis algorithms while providing a more complete harmonic analysis.

1. INTRODUCTION

Harmony, like rhythm, melody or timbre, is a central aspect of Western music. This paper focuses on chord sequences and key changes, which are strong components of the harmony. Audio chord transcription has been a very active field for the past recent years. In particular, the increasing popularity of Music Information Retrieval (MIR) with applications using mid-level tonal features, has established chord transcription as useful and challenging task.
Among the numerous chord recognition methods, we can distinguish four main types of systems. The first ones can be referred as *template-based methods* [7, 10, 15], since a central information they need to perform the transcription is the definition of the chords they want to detect. Working just like pattern recognition methods, they choose for every frame the chord whose template fits the best the data. The temporal structure of the song is often captured thanks to post-processing methods working either on the sequence of detected chords or on the calculated fitness features.
Other methods rely on musical information (such as rhythm or musical structure) in order to capture a harmonically relevant chord transcription. These *music-based methods* [2, 13], implicitly or explicitly exploit information from music theory in the construction of their systems. In particular, the transitions between chords or the rhythmic structure are often modeled with parameters reflecting musical knowledge, by estimating the likelihood of a given chord being followed by a different chord, for example. Some *data-driven methods* [11, 18], use completely or partially annotated data in order to build a system which fits the audio data. In these methods, all the parameters are evaluated with training. Finally, some systems merge music- and data-based approaches in order to build *hybrid methods* [16, 17], which combine the use of training data and music theory knowledge.

All these methods have the opportunity to compare to each other in the MIREX [5], which is an annual community-based framework for the evaluation of MIR systems and algorithms. In 2009, the results for the audio chord detection were pretty close and the different methods seemed to compete at the same level of accuracy. The aim of this to work is to offer a chord estimation with a comparable level of accuracy, and estimating a sequence of local keys as well as chords.

Fewer works were achieved to estimate musical keys from audio, and the vast majority of them only consider the main key (or global key) of a piece of music [9, 14]. In these works, because only the main key is handled, key changes are ignored (songs having different local keys are either ignored or considered to be in the first local key encountered). Chai [3] presented one of the few studies on key change from audio. In this work, local key tracking was performed by a HMM-based approach, and evaluated on ten classical piano pieces.

The main contribution of this paper relies in the fact that both chord and key can benefit from each other's estimation, as chords bring out information about local key and vice versa. We present a new system estimating simultaneously both chord and key sequences from audio. The proposed method is both *template-based* and *music-based* and no training is required.

We begin to present our work by describing the system used for both key and chord estimation in Section 2. Section 3 presents the experiments performed to evaluate the accuracy of the proposed method. Conclusion and future work follow in Section 4.

2. SYSTEM DESCRIPTION

In this section, we provide the description of the proposed method, which is adapted for audio from the proposed system in [anonymous self-reference]. The overall process is illustrated in Figure 1. The system works in four major steps: (1) chroma vectors are computed from audio signal; (2) a set of harmonic candidates are selected for each frame (Figure 1(a)); (3) a weighted acyclic graph of harmonic candidates is built (Figure 1(b)), (4) the dynamic process takes place (Figure 1(c)) and the final sequence

of chords/keys corresponding to the best path is outputted (Figure 1(d)).

An additional step consists in post-filtering the outputted sequence, to correct some analysis errors remaining.

2.1 Chroma Computation

The input audio file of the analysis system proposed is represented as sequences of chromas. This mid-level feature captures the tonal information since it represents the short-time energy related to each pitch class independently of octave [6]. Indeed, information about octave is not necessary for chord and key analysis purposes.

2.1.1 Tuning Issues

The chromas are computed on each frame. One of the main problem when analyzing audio musical piece is the variation in tuning. All the instruments are not always tuned to the same value, and this value often varies in time. Two options are possible. The tuning value may be analyzed, but tuning analysis assumes a stationarity. We choose to avoid this analysis by computing chroma on 36 bins and by shifting chroma at each frame according to possible tuning variations. This way, two chords played with different tuning result in two different 36 bin chromas, but results in almost the same 12 bin chroma [6].

2.1.2 Multi-Scale Approach

Instead of relying exclusively on one chromagram (sequence of chroma vectors over time), the proposed method includes a set of chromagrams. Each one has its own parameters but all share a common multiple hop size, to combine information at the same times during the piece of music. These chromagrams bring out different kinds of information, and may be subject to different treatments. Longer chromas may bring out information for key analysis, and different set of sizes for shorter chromas may fit different tempos and carry out different information useful for chord identification.

2.1.3 Filter

In order to reduce the influence of the noise, transients or sharp edges, we filter the chromagram on several frames [2, 7]. The filtering method used here is the median filtering, which has been widely used in image processing in order to correct random errors.

2.2 Selection of Harmonic Candidates

An harmonic candidate is a pair (C_i, K_i), where C_i (resp. K_i) represents a potential chord (resp. local key) for the ith frame of audio signal. C_i is then considered as a chord candidate (among possible others), and K_i as a key candidate. This section presents the processes allowing to select one or several chord/key pairs as harmonic candidate(s), and discard others.

2.2.1 Chord

The chords studied here are major and minor triads (12 major and 12 minors). Lots of works [7, 10, 15] have used chord templates to determine the likelihood of each of the 24 chords according to a chroma vector. With 12 dimensional vectors, major/minor triadic chord profile may be defined like the following:

$$\text{Major-triad} = (1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0)$$

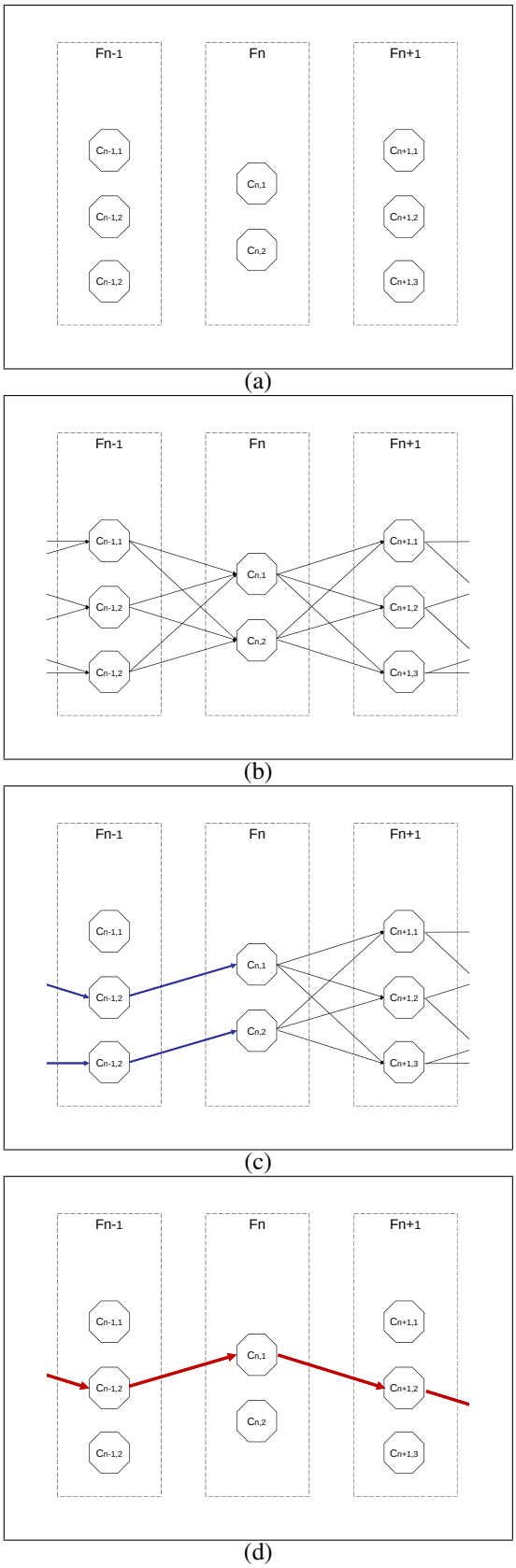

Figure 1. (a) Enumeration of harmonic candidates $C_{i,j}$ for consecutive audio frames F_i. $C_{i,j}$ represents the jth harmonic candidate for frame F_i. Time appears from left to right. (b) Creation of the edges of a weighted acyclic graph. An edge is built from each of the first frame's candidates to each of the second frame. (c) Dynamic process selects an unique path to each candidate of a given frame (here, frame n). (d) Selection of final path. The final chord/key sequences is then outputted from the sequence of chosen harmonic candidates.

$$\text{Minor-triad} = (1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0)$$

All the major (resp. minor) chord templates can be obtained by rotation of the major (resp. minor) profile.

For each of the 24 chord templates, we compute a correlation score by scalar product. The following details the correlation C between a chord template T and a 12 dimensional chroma vector V.

$$C_{T,V} = \sum_{i=1}^{12} (T[i].V[i])$$

The higher the correlation is, the more likely the chord corresponding to the template is played in the considered frame. A direct way to get chord candidates is thus by selecting the chords whose templates get the higher correlation score with the chroma of a given audio frame. In the multi-scale approach as exposed in Section 2.1.2, it is possible to consider different highest correlated chords from different windowed chromas as candidate for the same frame. The different chord candidates may thus be carrying different kind of information.

2.2.2 Key

Key selection is carried out with the same approach as for chords, but with larger time frame as keys have a larger time persistence than chords. The key profiles used are presented in [19]:

$$\text{Major} = (5, 2, 3.5, 2, 4.5, 4, 2, 4.5, 2, 3.5, 1.5, 4)$$

$$\text{Minor} = (5, 2, 3.5, 4.5, 2, 4, 2, 4.5, 3.5, 2, 1.5, 4)$$

As for chord candidate computation, the correlation of each of the 24 keys (12 minors + 12 majors) are computed using a scalar product between shifted key template and chroma vectors.

2.2.3 Harmonic candidates

The harmonic candidates finally enumerated are all the possible combination of previously selected keys and chords. If n chords and m keys are selected for a given audio frame, n x m pairs are enumerated. For example, with C_M and A_m as selected chords and C_M and G_M as compatible keys, the harmonic candidates enumerated would be (C_M, C_M), (C_M, G_M), (A_m, C_M) and (A_m, G_M). A different choice can be made, by considering a compatibility between chords and keys. But an incorrect chord selected may discard the correct key (and vice versa), because the two are not compatible. For this reason, adding a compatibility between chords and keys has led to a decrease of accuracy.

2.3 Weighted Acyclic Harmonic Graph

Once the harmonic candidates are enumerated for two consecutive frames, an edge is built from each of the first frame's candidates to each of the second frame. This edge is weighted by a transition cost between the two chord candidates. This transition cost must take into account both the different selected chords, and the different selected local keys.

We thus choose to use Lerdahl's distance [12] as a first transition cost.

This distance is based on the notion of basic space. Lerdahl defines the basic space of a given chord in a given key as the geometrical superposition of:
(a) the chromatic pitches of the given key (chromatic level),

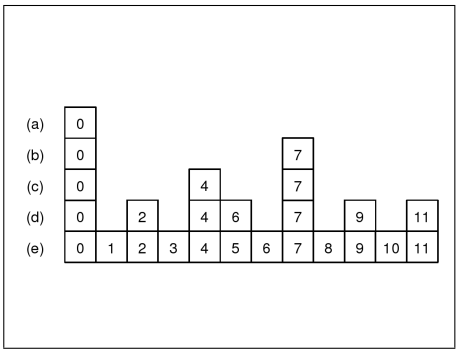

Figure 2. The basic space of the C_M chord in the C_M key. Levels (a) to (e) are respectively chromatic, diatonic, triadic, fifths and root levels.

(b) the diatonic pitches of the given key (diatonic level),
(c) the triad pitches of the given chord (triadic level),
(d) the root and dominant of the given chord (fifths level),
(e) the root of the given chord (root level).

Figure 2 shows the basic space of the C_M chord in the C_M key.

If (C_x, K_x) represents the chord C_x in the key K_x, Lerdahl defines the transition cost from $x = (C_x, K_x)$ to $y = (C_y, K_y)$ as follows:

$$\delta(x \to y) = i + j + k$$

where i is the distance between K_x and K_y in the circle of fifths, j is the distance between C_x and C_y in the circle of fifths and k is the number of non-common pitch classes in the basic space of y compared to those in the basic space of x (see [12] for more details).

The distance thus provides an integer cost from 0 to 13, and is adequate for a transition cost in the proposed method, since both compatible chords and keys are involved in the cost computation. Nevertheless, this distance offers a small range of possible values. As we need to compare different paths between harmonic candidates, this small range induces a lot of equality scenarios. The Lerdahl's distance is thus slightly modified and the cost between two consecutive candidate is set to $i^\alpha + j^\beta + k$, with i, j and k defined in Section 2.3. We choose $\alpha > 1$ to discourage immediate transitions between distant keys, and encourage progressive key changes, since modulations often involve two keys close to each other in the circle of fifths. For the same reason with chords, we also choose $\beta > 1$. After experiment, α and β have been set to 1.1 and 1.01.

A calculation of chord transition is illustrated in Figure 3, from $x = (C_M, C_M)$ to $y = (G_M, G_M)$. Here, i=j=1 because 1 step is needed to go from C_M to G_M in the circle of fifths. k=5 is the number of non-common pitches belonging to the basic space of of y compared to those in the basic space of x (underlined in the Figure). The distance is therefore 1+1+5=7.

The Lerdahl's distance is then combined with a second transition cost, derived from a chord transition matrix. This 24x24 triad transition matrix was created as follows. The Bas De Haas corpus [4], which contains 5000 chord sequences, was first converted to a simpler triadic representation using simple transformation rules. Runs of the same

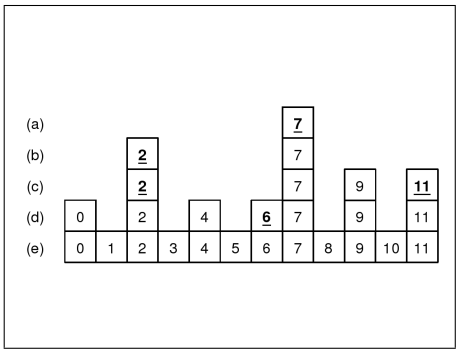

Figure 3. $\delta((C_M, C_M) \rightarrow (G_M, G_M)) = i + j + k = 1 + 1 + 5 = 7$. The underlined pitches are the non-common pitches.

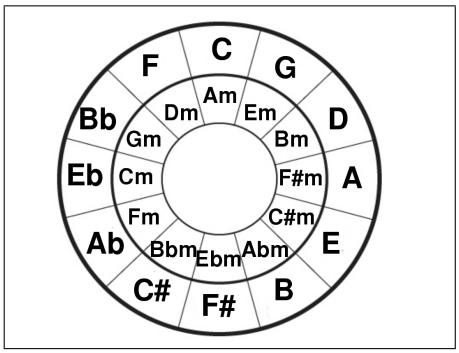

Figure 4. The circle of fifths.

triad were then collapsed to a single occurrence. The derived sequences were then viewed as sequences of relative transitions (maj:maj, maj:min, min:min, and min:maj) linked with root movement interval modulo 12. Frequencies of all 48 values were compiled directly from the chord corpus, leading to two 1x24 transition vectors (one for transitions out of major triads, one from minor triads). To produce a final 24x24 transition matrix indexed by triad, these compiled frequencies were then replicated 12 times. To obtain a transition cost $\delta(x \rightarrow y)$ from the transition matrix, we apply the following formula:

$$\delta(x \rightarrow y) = -log(M[i][j])$$

where i is the row corresponding to the x chord, and j is the column corresponding to the y chord. When $x = y$, the transition cost derived from the matrix is set to 0.

2.4 Finding the Best Path

Once the graph between all the harmonic candidates is formed, the best path has to be found. This task is achieved by dynamic programming [1]. In the graph, from left to right, only one edge to each harmonic candidate is preserved. Several ways to select this edge can be considered. We choose to preserve the edge minimizing the total sum of weights along the path leading to each candidate, as illustrated in Figure 1(c). The number of final paths is the number of harmonic candidates for the last frame. The final selected path is the path minimizing its total cost along its edges. This path is outputted by the program.

2.5 Post-smoothing computation

Among the selected sequence of chord/key, some errors may still be corrected by applying a post-smoothing treatment. For example, if an instrument (or a singer) plays a flattened third (Eb) as a blue note, it may induce a mode error on the selected chord (making C_m as a chord candidate and discarding C_M for the considered frame). The outputted chord sequence may thus present several consecutive frames analyzed as C_M are followed by a single frame analyzed as C_m, and then by another several C_M. A simple post treatment on the outputted sequence of chords may resolve this kind of errors.

3. EXPERIMENTS

This section presents the database used for experiments, the evaluation procedure, and the influence on the different parameters on the system accuracy. Once the best settings determined, we compare the system to a state-of-the-art method for chord estimation, and a direct template-based method for key estimation.

3.1 Database

As both local key and chord ground truth were needed, we choose to evaluate the proposed system on the Beatles audio discography (174 songs) with a 44100 Hz sampling rate. In this database, the average number of chord changes by song is 69, with an average of 7.7 different chords by song. The average number of different local keys by song is 1.69. Chords transcriptions were checked by Christopher Harte and the MIR community, and keys annotations were provided by the Centre for Digital Music (C4DM). Both sets of transcriptions are available at `http://www.isophonics.net`.

3.2 Evaluation

In the transcriptions, chords have a root note and a type which belongs to a vast dictionary [8]. In this paper, we only focus on the root note (C, C#, D, ..., B) and the mode (maj/min) of chords. All the ground truth chords of the database have thus been mapped to min/maj triads. When the chord has no third and cannot be mapped to a min/maj triad, only the root note is considered and later compared to the corresponding estimated root note. Silences and no-chords (part of a song when no chord is defined) are ignored, as the chord/no-chord detection issue has not yet been addressed in the proposed system. The other components of the evaluation are the same as for the evaluation led in the 2009 MIREX audio chord detection task [1]. The audio signal is divided in frames of approximately 100 ms (4096 audio samples). The estimated chord is compared for each frame to the ground truth at the time corresponding to the center of the frame. The final score for a song is the number of frames where estimated chord matches the ground truth divided by the number of frames analyzed. For the local key evaluation, the procedure is identical. For each frame, the estimated key is compared to the ground truth key at the center of the frame.

[1] `http://www.music-ir.org/mirex/2009/index.php/Audio_Chord_Detection_Results`

No filtering

Nb. of chord candidates	1	2	3	4
Ratio of correct. (%)	58.3	71.5	78.6	82.9
System (%)	58.3	**64.9**	64.1	62.2

Filtering

Nb. of chord candidates	1	2	3	4
Ratio of correct. (%)	68.4	79.1	85.5	88.9
System (%)	68.4	**70.0**	64.3	59.3

Table 1. Percentage of correct chord in harmonic candidates and system output accuracy depending on the number of candidates and chroma filtering. Best results are achieved by limiting the number of candidates and filtering chromas.

3.3 Chord Estimation

Following the multi-scale approach presented in 2.1.2, we need to use different size of chromas vectors for chord estimation. The parameters for the different chroma scales are the following:

- "long" chromas: 32768 samples as window length (approximately 0.8 sec) and 8192 (approximately 0.2 sec) as hop size,
- "medium" chromas: 8192 samples as window length (approximately 0.2 sec) and 8192 (approximately 0.2 sec) as hop size,
- "short" chromas: 4096 samples as window length (approximately 0.1 sec) and 4096 (approximately 0.1 sec) as hop size.

3.3.1 Influence of Filtering

A first experiment has been carried out on long chromas to measure the influence of chroma filtering on chord estimation. For each frame, we set as chord candidate the n highest correlated chords with the maj/min chord templates . Tests go from $n = 1$ to $n = 4$. For each value of n, we compute the ratio of correctness as the number of frames for which the correct chord is among the selected candidates over the total number of frames. This ratio represents the system theoretical maximum accuracy, and is reached if every correct chord candidate is present in the final chord sequence outputted. Obviously, the higher the number of considered chord candidates is, the higher the chance is for one of them to be the correct chord, and thus the higher the ratio of correctness is. In the other hand, more chord candidates considered increase the likelihood of the system to pick a incorrect chord. Finding the balance between these two parameters is thus a real need. Table 1 presents the ratio of correctness and the system score with or without chroma filtering and for different values of n (number of chord candidates). Best chroma filtering setting is achieved by taking into account a window of 9 chromas, centered on the considered chroma.

These first results shows that filtering leads to an improvement of the ratio of correctness, from 6% with four selected candidates (82.9% to 88.9%) to more than 10% with one (58.3% to 68.4%). Filtering thus seems to correct some errors due to chroma vectors, by taking into account information from adjacent frames.

Tri-candidate (1 long and 2 short chromas)

Filtering	none	long	short	both
Ratio of correct. (%)	69.8	78.2	76.6	79.1
Distinct chords (av.)	1.86	1.95	1.43	1.36
System (%)	64.5	72.2	71.8	**73.7**

Bi-candidate (1 long and 1 medium chromas)

Filtering	none	long	medium	both
Ratio of correct. (%)	65.1	75.1	73.5	76.7
Distinct chords (av.)	1.38	1.46	1.29	1.23
System (%)	62.8	71.6	70.7	**72.8**

Table 2. Percentage of correct chord in harmonic candidates and system score depending on the selection of candidates from different sized chromagrams. Each time, the average number of distinct chord candidates is mentioned. Best results are reached by filtering both long and short chromagrams, and by combining their information. Tri-candidate means the two best candidates from the two adjacent short chromas centered in a long chroma with the best candidate from the long chroma. Bi-candidate means the best candidates from the medium chroma centered in a long chroma with the best candidate from the long chroma.

3.3.2 Influence of the Number of Chord Candidates

In Table 1, we notice the drop of the system's performance when the number of selected candidates per frame exceeds two. This can be explained by the close relationship existing among the highest correlated chord candidate of a given chroma vector. Indeed, chord templates of two major and minor chords sharing the same root note often induce a close correlation score for a given chroma. The same goes for any couple of chords close to each other in terms of Lerdahl's distance. In 80% of the frames, top 2 correlated chord candidate have a distance less or equal to 1 on the circle of fifths. Considering different candidates from the same chroma thus does not seem profitable to gain a maximum system accuracy.

3.3.3 Influence of the Multi-Scale Approach

Since a drop of accuracy is noticed when too many candidates from the same chroma are selected as candidates, we propose a new approach by considering top correlated candidates from different sized chromas. Table 2 presents the ratio of correctness as well as the system score depending on the combination of chroma size, and filtering. The general idea is to add highest correlated chord candidates from shorter chromas to the highest chord candidate of a given long chroma. Tri-candidate means the combination of the two best candidates from the two adjacent short chromas centered in a long chroma with the best candidate from the long chroma. Bi-candidate means the combination of the best candidate from the medium chroma centered in a long chroma with the best candidate from the long chroma. Since top correlated chords of two different sized chroma may be identical, we also show the average number of distinct chord candidates per frame. As for the previous experiment, best filtering is achieved by taking into account a windows of 9 chromas and centered on the considered chroma, whatever its length.

Filtering effective in each case. If filtering is applied to only one scale, the system accuracy and ratio of correct-

Method	Root	Root and Mode
OGF2 (%)	**78.9**	72.3
Proposed System (%)	77.9	**74.9**

Table 3. Comparison of the proposed method with the OGF2 method, which scored 1st (resp 2nd) in the 2009 MIREX Audio Chord Detection "root estimation" task (resp. root and mode task).

ness both benefit a little more of the long chroma filtering than the short (resp. medium) for the tri-candidate (resp. bi-candidate). Nevertheless, the best overall filtering efficiency is reached by combining the filter on both chromas size. Compared to no filter, the double sized-filter leads to an increase of around 10% for the ratio of correctness and the system accuracy, both in the tri-candidate and the bi-candidate cases. The difference between the ratio of correctness and the system score is less important by considering candidates from different chromas than by considering several candidates from the same chroma. With filtering, this difference is of 5.4% (79.1 - 73.7) for the tri-candidate configuration and of 2.9% (76.7 - 72.8) for the bi-candidate (see Table 2), when it is more than 9% (79.1 - 70) with 2 candidates from the same long chromas (see Table 1). A first way to explain this improvement is by considering the decrease of average number of distinct chord candidates, which is always lesser than 2 in both tri-candidate and bi-candidate configuration. This decrease means fewer chord candidates to consider for the system, thus decreasing the likelihood to select an incorrect chord.

Maximum accuracy is reached with tri-candidate configuration, with a system accuracy of 73.7%.

3.3.4 Post-Smoothing Treatment

We decide to apply a post-smoothing treatment to the output of the system with the best settings, which performs a 73.7% accuracy (see Table 2). The post-smoothing filter looks for output chord sequence in the form of $...AABAA...$ (resp. $...AAABCAAA...$) and corrects it in $...AAAAA...$ (resp. $...AAAAAAAA...$). By applying the post-smoothing treatment with the system best previous settings, chord detection reaches a **74.9%** accuracy.

3.3.5 Comparison to a state-of-the-art Method

We compare the best configuration of our system to one of the best methods of the 2009 MIREX Audio Chord Estimation task, evaluated on the same database with the same evaluation procedure. The comparison is made with the OGF2 method, proposed by Oudre et al. Results are shown in Table 3. On the root estimation only, OGF2 is 1% more accurate than the proposed method (78.9% compared to 77.9%). On the root and mode estimation, the proposed system performs better than the OGF2 method and improves by almost 3% the accuracy of the detected chords (74.9% compared to 72.3%). This comparison shows that the proposed method is comparable, and maybe even more accurate than one of the best methods presented at the 2009 MIREX when it comes to chord estimation, and compares the local key sequence as well as the chord sequence.

3.4 Local Key Estimation

Key estimation is performed on the same database than for chord estimation. We compare the key sequence output of the proposed system to a direct template-based method (DTBM). The same settings are used for the two compared methods, as we wish to evaluate the system's contribution. The window size is set to 30 sec approximately. For the proposed method, the number of key candidates per frame is set to 3. Results, shown in Table 4, detail the estimated key error made by the two compared method, by presenting relative and neighbor errors as well as correct key accuracy. Relative keys share the same key signature (for example, C_M and A_m are relative keys of each other). A neighbor key differs from the original key by an accidental. Each key has two neighbors (for instance, C_M has F_M and G_M as neighbors).

Key estimation	Correct	Rel.	Nei.	Oth.
System (%)	**62.4**	2.9	17.4	**17.3**
DTBM (%)	57.6	1.6	18.9	21.9

Table 4. DTBM: Direct Template-Based Method. Keys scores are shown in % and are split among possible errors: correct keys, relative keys (Rel.), neighbor keys (Nei.) and others (Oth.). The system performs the highest accuracy in terms of correct key detected. The number of due to non related keys is also less important with the system analysis than with the DTBM.

Harmonic Candidate	Chord C	Chord •	Key •	Key K	Both C	Both K
System (%)	73.1	•	•	57.8	**74.9**	**62.4**

Table 5. System accuracy considering (chord,key), only chord and only key as harmonic candidates. The system performs better in chord and key estimation when taking into account both information simultaneously.

The system performs better than the DTBM method by estimating more correct keys (62.4% compared to 57.6%). Moreover, the number of errors due to non related key (different from neighbor or relative) is less important when the analysis is performed by the system (17.3% compared to 21.9%).

3.5 Reciprocal Benefit of Simultaneous Estimation

We present here an evaluation to measure the reciprocal influence of the chord and key simultaneous estimation. We compared the proposed system, which takes into account harmonic candidates (i.e. pairs of chord AND key candidates), to the same system with only chord OR key candidates. When only chord (resp.) are considered, the distance used to weigh edges in the harmonic graph is edited to take only chord (resp. key) into account. Results are shown in Table 5.

We note that both key and chord estimation are better when the harmonic candidate is the (chord,key) pair. Chord estimation accuracy drops of almost 2% (74.9 compared to 73.1) and key estimation accuracy drops of almost 5% (62.4 compared to 57.8).

4. CONCLUSION AND FUTURE WORK

This paper presents a new method for chord and local key estimation where the analysis of chord sequence and key changes are performed simultaneously. A multi-scale ap-

proach for chroma vectors is proposed, and we show an increase in accuracy when the chords are selected from different sized chromas. While the key estimation performs better than a direct template-based method, the chord accuracy shows better results than a state-of-the-art method.

Future work will involve analysis of different chord types, silence and no-chord detection as well weighing the harmonic graph of the proposed method in a probabilistic approach. Applications for MIR using both local key and chord information are also studied. For example, harmonic information may be helpful for estimating the musical structure of pieces since changes of local key generally occur at the beginning of new patterns. Furthermore, we aim at investigating the possible improvements induced by a retrieval system based on all the harmonic information, compared to existing systems that only consider chord progressions.

5. REFERENCES

[1] R. Bellman. *Dynamic Programming*. Princeton University Press, 1957.

[2] J.P. Bello and J. Pickens. A robust mid-level representation for harmonic content in music signals. In *Proc. of the International Symposium on Music Information Retrieval (ISMIR)*, pages 304–311, London, UK, 2005.

[3] W. Chai and B. Vercoe. Detection of key change in classical piano music. In *Proc. of the International Symposium on Music Information Retrieval (ISMIR)*, pages 468–473, London, UK, 2005.

[4] W. Bas De Haas, Matthias Robine, Pierre Hanna, Remco C. Veltkamp, and Frans Wiering. Comparing harmonic similarity measures. In *Proceedings of the 7th International Symposium on Computer Music Modeling and Retrieval (CMMR)*, pages 299–315, June 2010.

[5] J. Stephen Downie. The music information retrieval evaluation exchange (2005–2007): A window into music information retrieval research. *Acoustical Science and Technology*, 29(4):247–255, 2008.

[6] E. Gómez. *Tonal Description of Music Audio Signals*. PhD thesis, University Pompeu Fabra, Barcelona, Spain, July 2006.

[7] C. Harte and M. Sandler. Automatic chord identification using a quantised chromagram. In *Proc. of the Audio Engineering Society*, Barcelona, Spain, 2005.

[8] C. Harte, M. Sandler, and A. Samer. Symbolic representation of musical chords: A proposed syntax for text annotations. In *Proc. 4th Int. Conf. on Music Information Retrieval (ISMIR), 2005*, pages 66–71, 2005.

[9] O. Izmirli. Audio key finding using low-dimensional spaces. In *Proc. of the International Symposium on Music Information Retrieval (ISMIR)*, pages 127–132, Victoria, Canada, 2006.

[10] K. Lee. Automatic chord recognition from audio using enhanced pitch class profile. In *Proc. of the International Symposium on Music Information Retrieval (ISMIR)*, Victoria, Canada, 2006.

[11] K. Lee and M. Slaney. Acoustic chord transcription and key extraction from audio using key-dependent HMMs trained on synthesized audio. *IEEE Trans. on Audio, Speech and Language Processing*, 16(2):291–301, 2008.

[12] F. Lerdahl. *Tonal Pitch Space*. Oxford University Press, 2001.

[13] M. Mauch and S. Dixon. Simultaneous estimation of chords and musical context from audio. *IEEE Trans. on Audio, Speech and Language Processing*, 2010.

[14] K. Noland and M. Sandler. Key estimation using a hidden markov model. In *Proc. of the International Symposium on Music Information Retrieval (ISMIR)*, pages 121–126, Victoria, Canada, 2006.

[15] L. Oudre, Y. Grenier, and C. Févotte. Template-based chord recognition : influence of the chord types. In *Proc. of the International Symposium on Music Information Retrieval (ISMIR)*, pages 153–158, Kobe, Japan, 2009.

[16] H. Papadopoulos and G. Peeters. Large-scale study of chord estimation algorithms based on chroma representation and HMM. In *Proc. of the International Workshop on Content-Based Multimedia Indexing*, pages 53–60, Bordeaux, France, 2007.

[17] M.P. Ryynänen and A.P. Klapuri. Automatic transcription of melody, bass line, and chords in polyphonic music. *Computer Music Journal*, 32(3):72–86, 2008.

[18] A. Sheh and D.P.W. Ellis. Chord segmentation and recognition using EM-trained hidden Markov models. In *Proc. of the International Symposium on Music Information Retrieval (ISMIR)*, pages 185–191, Baltimore, MD, 2003.

[19] D. Temperley. *The Cognition of Basic Musical Structures*. The MIT Press, 1999.