Chord recognition by fitting rescaled chroma vectors to chord templates

Transcription automatique en accords par minimisation de mesures entre vecteurs de chroma et modèles d'accord

Laurent Oudre
Yves Grenier
Cédric Févotte

2009D019

octobre 2009

Département Traitement du Signal et des Images
Groupe AAO : Audio, Acoustique et Ondes

Technical report

Chord recognition by fitting rescaled chroma vectors to chord templates

Rapport technique

Transcription automatique en accords par minimisation de mesures entre vecteurs de chroma et modèles d'accords

Laurent Oudre, Yves Grenier, Cédric Févotte

Institut TELECOM, TELECOM ParisTech, CNRS LTCI
46 rue Barrault, 75634 Paris Cedex 13, France

october/octobre 2009

Contents

1 Introduction **4**

2 State of the art **6**

3 Our System **8**
3.1 General idea 8
3.2 Chord models 8
3.3 Measures of fit 9
3.4 Filtering methods 10

4 Evaluation and corpus **12**
4.1 Beatles corpus 12
4.2 MIDI corpus 13
4.3 Evaluation method 13
4.4 Input features 14

5 Results on the Beatles corpus **16**
5.1 Results with major/minor chord types 16
5.2 Results with other chord types 17
5.3 Comparison with the state-of-the-art 18
5.4 Analysis of the errors 18

6 Results on the MIDI corpus **21**
6.1 Influence of the music genre 21
6.2 Influence of the percussive noise 22
6.3 Beat synchronous chord detection 22

7 Conclusion **23**

Abstract

In this paper we propose a simple and fast method for chord recognition in music signals. We extract a chromagram from the signal which transcribes the harmonic content of the piece over time. We introduce a set of chord templates taking into account one or more harmonics of the pitch notes of the chord and calculate a scale parameter to fit the chromagram frames to these chords templates. Several types of chords (major, minor, dominant seventh,...) are considered. The detected chord over a frame is the one minimizing a measure of fit between the rescaled chroma vector and the chord templates. Several popular distances and divergences from the signal processing or probability fields are considered for our task. Our system is improved by some post-processing filtering that modifies the recognition criteria so as to favor time-persistence.

The transcription tool is evaluated both on the Beatles corpus used for MIREX 08 and a resynthesized MIDI corpus. Our system is also compared to state-of-the-art chord recognition methods. Experimental results show that our method outperforms the state-of-the-art and more importantly is less computationally demanding than the other evaluated systems.

Index terms : chord recognition, music signal processing, music signal representation, music information retrieval

Résumé

Nous proposons ici une méthode simple et rapide pour la reconnaissance d'accords dans les signaux musicaux. On extrait d'abord du signal un chromagramme qui traduit le contenu harmonique du morceau en fonction du temps. On introduit un ensemble de modèles d'accords qui tiennent en compte d'une ou plusieurs harmoniques des notes de l'accord, et on calcule un paramètre d'échelle afin d'adapter les trames de chromagramme à ces modèles d'accords. Plusieurs types d'accords (majeur, mineur, septième,...) sont considérés. L'accord détecté pour une trame est celui minimisant une mesure entre le vecteur de chroma mis à l'échelle et le modèle d'accord. Plusieurs distances et divergences célèbres dans le domaine du traitement du signal et des probabilités sont considérées pour notre tâche. Notre système est ensuite amélioré grâce à des méthodes de post-traitement qui modifient les critères de reconnaissance pour prendre en compte la persistance temporelle.

Cet outil de transcription est évalué sur deux corpus : un corpus constitué par l'intégralité des titres des Beatles (déjà utilisé pour MIREX 08) et un corpus de fichiers MIDI resynthétisés. Notre système est aussi comparé à l'état de l'art. Les résultats expérimentaux montrent que notre méthode dépasse l'état de l'art et est surtout plus rapide que les autres systèmes évalués.

Mots clés : reconnaissance d'accords, traitement du signal musical, recherche

d'information musicale

1 Introduction

Complete musical analysis of a pop song, that is to say the transcription of every single note played by every instrument is a very complex task. The musical content of a pop song is thus more often translated into a more compact form such as sequences of chords. A chord is a set of notes played simultaneously. A chord can be defined by a *root note* which is the note upon which the chord is perceived and a *type* giving the harmonic structure of the chord. For example a *C major* chord is defined by a root note C and a type *major* which indicates that the chord will also contain the major third and the perfect fifth, namely the notes E and G. The result of chord transcription consists in sequences of chords played successively with their respective lengths. This compact and robust writing not only helps to play-back the song but also gives information on the harmonic content and structure of the song. Automatic chord transcription finds many applications in the field of Musical Information Retrieval. The characterization of a song by its chord transcription can be used in several tasks among which song identification, query by similarity or analysis of the structure of the piece.

Automatic chord transcription includes in most cases two successive steps : a feature extraction which captures the musical information and a recognition process which outputs chord labels from the extracted features.

The first step consists in the extraction of relevant and exploitable musical content from the audio. As such, pitch perception of a note can be decomposed into two different notions : *height*, corresponding to the octave to which the note belongs and *chroma* or *pitch class* indicating the relation of the note with the other notes among an octave. For example the note $A4$ (440 Hz) is decomposed into an octave number 4 and a chroma A. The features used in chord transcription may differ from a method to another but are in most cases variants of the *Pitch Class Profiles* introduced by Fujishima [1] whose calculation is based on this notion of chroma. These features, also called *chroma vectors*, are 12-dimensional vectors. Every component represents the spectral energy of a semi-tone on the chromatic scale regardless of the octave. These features are widely used both in chord recognition and tonality extraction. The calculation is based either on the *Constant Q Transform (CQT)* [2] or on the *Short Time Fourier Transform (STFT)* and is performed either on fixed-length frames or variable-length frames (depending for example on the tempo, etc.). The succession of these chroma vectors over time is often called *chromagram* and gives a good representation of the musical content of a piece.

The structure of a chord being entirely defined by its root note and type, it is easy to create 12-dimensional chord templates which reflect this structure by giving a particular amplitude to every chroma. The simplest model for chords, widely used in chord recognition [1], [3], has a binary structure giving an amplitude of 1 to the chromas constituting the chord and 0 for the other chromas. Other models can be introduced for example by taking into account

the harmonics of the notes played in the chord [4], [5].

The present paper focuses mainly on the second part of the chord transcription process that is to say the chord labeling of every chromagram frame. Our chord recognition system is based on the intuitive idea that for a given 12-dimensional chroma vector, the amplitudes of the chromas present in the chord played should be larger than the ones of the non-played chromas. By introducing chord templates for different chord types and roots, the chord present on a frame should therefore be the one whose template is the *closest* to the chroma vector according to a specific measure of fit. A scale parameter is introduced in order to account for amplitude variations and finally the detected chord is the one minimizing the measure of fit between the rescaled chroma vector and the chord templates.

Section 2 provides a review of the state-of-the-art methods for the chord recognition. Section 3 gives a description of our recognition system : the chord templates, the measures of fit and some post-processing filtering methods exploiting time-persistence. Section 4 describes the evaluation protocol for our method. Section 5 presents a qualitative and quantitative analysis of the results on a data corpus formed by the 13 Beatles albums and a comparison with the state-of-the-art. Section 6 gives results on another corpus composed of audio files synthesized from MIDI and investigates the influence of the genre, percussive noise and beat-synchronous chord detection.

2 State of the art

The chord recognition task consists in outputting a chord label from a specific music-related feature. Most chord recognition systems use a chromagram (or assimilate) as an input to the system and output a chord label for each chromagram frame. Machine-learning methods such as Hidden Markov Models (HMMs) have been widely used for this task especially in the last years but templates-fitting techniques have also been used for this labeling process.

A Hidden Markov Model is constituted by a number of hidden states with an initial state distribution, a state transition probability distribution which gives the probability of switching from a state to another and an observation probability distribution which gives the likelihood of a particular state for a particular observation data. In the typical HMM-based chord recognition systems every chord is represented by a hidden state and the observations are the chromagram frames. Given the parameters of the model, the chord recognition consists in finding the most likely sequence of hidden states (chords) that could have generated a given output sequence (chromagram). The parameters of these HMMs (initial state distribution, state transition probability distribution and observation probability distributions) are either based on musical theory, learned on real data or a combination of these two approaches.

The first HMM used in chord recognition [6] is composed of 147 hidden states each representing a chord and corresponding to 7 types of chords (major, minor, dominant seventh, major seventh, minor seventh, augmented and diminished) and 21 root notes (12 semi-tones with the distinction between $\flat$ and $\sharp$). All the HMM parameters are learned by a semi-supervised training with an EM algorithm. This model is then improved in [7] by a complete re-building of the HMM. The number of hidden states is reduced from 147 to 24 by only considering major and minor chords ; this enables to have sufficient data for the training process. The initializations for the HMMs parameters are inspired by musical and cognitive theory which naturally introduced musical knowledge into the model. The state transition probability distribution and the initial state distribution are still updated by an unsupervised training with an EM algorithm but the observation probability distributions are fixed, giving to each chord a clear and predetermined structure. The introduction of tempo-based features also enhances the recognition performances. Some other methods [5], [8] also use a 24 states HMM considering only major and minor chords but try different sets of input features, HMM parameters or training approaches. Symbolic data can be used for the training process with a system based on 24 tonality-dependent HMMs [9] in order to give a joint key extraction and chord transcription.

Yet, the first chord recognition system based on chroma representation proposed by Fujishima [1] is not using HMM but chord dictionaries composed of 12-dimensional templates constituted by 1 (for the chromas present in the chord) and 0 (for the other chromas). 27 types of chords are tested and the transcription is done either by minimizing the Euclidean

distance between *Pitch Class Profiles* and chord templates or by maximizing a weighted dot product. Fujishima's system is improved [3] by calculating a more elaborate chromagram including notably a tuning algorithm and by reducing the number of chords types from 27 to 4 (major, minor, augmented, diminished). Chord transcription is then realized by retaining the chord with higher dot product between the chord templates and the chromagram frames. Chord transcription can also be done by maximizing the correlation between enhanced variants of the *Pitch Class Profiles* and chord templates [10]. These chord templates are also used on MIDI data for the joint tasks of segmentation and chord recognition [11] by the calculation of weights reflecting the similarity between the chord models and the present notes in a segment.

3 Our System

3.1 General idea

Let $\mathbf{C}$ denote the chromagram, with dimensions $M \times N$ (in practice $M = 12$) composed of N successive chroma vectors $\mathbf{c}_n$. Let $\mathbf{p}_k$ be the 12-dimensional chord template defining chord k. We want to find the chord k whose template $\mathbf{p}_k$ is the *closest* to the chromagram frame $\mathbf{c}_n$ for a specific measure of fit. We propose to measure the fit of chroma vector $\mathbf{c}_n$ to template $\mathbf{p}_k$ up to a scale parameter $h_{k,n}$. Given a measure $D(\,.\,;\,.\,)$, a chroma vector $\mathbf{c}_n$ and a chord template $\mathbf{p}_k$, the scale parameter $h_{k,n}$ is calculated analytically to minimize the measure between $h\,\mathbf{c}_n$ and $\mathbf{p}_k$:

$$h_{k,n} = \underset{h}{\operatorname{argmin}}\ D\left(h\,\mathbf{c}_n; \mathbf{p}_k\right). \tag{3.1}$$

In practice $h_{k,n}$ is calculated such that :

$$\left[\frac{d\,D\left(h\,\mathbf{c}_n; \mathbf{p}_k\right)}{dh}\right]_{h=h_{k,n}} = 0. \tag{3.2}$$

We then define $d_{k,n}$ as :

$$d_{k,n} = D\left(h_{k,n}\,\mathbf{c}_n; \mathbf{p}_k\right). \tag{3.3}$$

The detected chord $\hat{k}_n$ for frame n is then the one minimizing the set $\{d_{k,n}\}_k$:

$$\hat{k}_n = \underset{k}{\operatorname{argmin}}\ d_{k,n}. \tag{3.4}$$

3.2 Chord models

The chord templates are 12-dimensional vectors where each component represents the theoretical amplitude of each chroma in the chord. These chord templates can either be learned on audio data [6], [8], [9] or predetermined [1], [3], [5], [7], [10], [11]. However, Bello & Pickens [7] and Papadopoulos & Peeters [5] have shown that using fixed and musically inspired chord structures can give better results for the chord detection task. Besides, the use of fixed chord models allows to skip the time-consuming learning phase and the need of annotated training data.

In our system three chord models are defined : examples for C major and C minor chords are displayed on Figure 3.1.

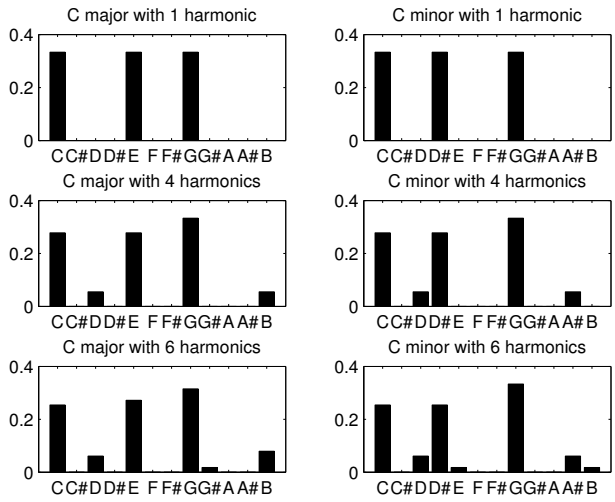

Figure 3.1: Chord templates for C major / C minor with 1, 4 or 6 harmonics.

The **first chord model** is a simple binary mask : an amplitude of 1 is given to the chromas defining the chord and an amplitude of 0 is given to the other chromas.[1] For example for a C *major* chord an amplitude of 1 is given to the chromas C, E and G while the other chromas have an amplitude of 0.

The **second chord model** is inspired from the work of Gomez [4] and Papadopoulos [5]. The information contained in a chromagram or any other spectral representation of a musical signal captures not only the intensity of every note but a blend of intensities for the harmonics of every note. It is therefore interesting and relevant to take into account the harmonics for each note of the played chord. An exponentially decreasing spectral profile is assumed for the amplitudes of the partials and an amplitude of s^{i-1} is added for the i^{th} harmonic of every note in the chord. The parameter s is empirically set to 0.6. Our second chord model only takes into account the 4 first harmonics.

The **third chord model** is based on the same principle but takes into account the first 6 harmonics for the notes of the chord.

From these three chord models we can build chord templates for all types of chords (major, minor, dominant seventh, diminished, augmented,...). By convention in our system, the chord templates are normalized so that the sum of the amplitudes is 1 but any other normalization could be employed.

3.3 Measures of fit

We consider for our recognition task several measures of fit, popular in the field of signal processing. Table 3.1 gives the expressions of these different measures, as well as the scale parameter analytically calculated from (3.2) and the final expression of the set of values $d_{k,n}$.

[1] In practice a small value is used instead of 0, to avoid numerical instabilities that may arise with some measures of fit, see section 3.3.

	Expression of $D\left(h_{k,n}\,\mathbf{c}_n;\mathbf{p}_k\right)$	Scale parameter $h_{k,n}$	Minimization criteria $d_{k,n}$
EUC	$\sqrt{\sum_m \left(h_{k,n}\,c_{m,n}-p_{m,k}\right)^2}$	$\frac{\sum_m c_{m,n}\,p_{m,k}}{\sum_m c_{m,n}^2}$	$\sqrt{\sum_m p_{m,k}^2-\frac{\left(\sum_m c_{m,n}\,p_{m,k}\right)^2}{\sum_m c_{m,n}^2}}$
IS1	$\sum_m \frac{h_{k,n}\,c_{m,n}}{p_{m,k}}-\log\left(\frac{h_{k,n}\,c_{m,n}}{p_{m,k}}\right)-1$	$\frac{M}{\sum_m \frac{c_{m,n}}{p_{m,k}}}$	$M\log\left(\frac{1}{M}\sum_m \frac{c_{m,n}}{p_{m,k}}\right)-\sum_m \log\left(\frac{c_{m,n}}{p_{m,k}}\right)$
IS2	$\sum_m \frac{p_{m,k}}{h_{k,n}\,c_{m,n}}-\log\left(\frac{p_{m,k}}{h_{k,n}\,c_{m,n}}\right)-1$	$\frac{1}{M}\sum_m \frac{p_{m,k}}{c_{m,n}}$	$M\log\left(\frac{1}{M}\sum_m \frac{p_{m,k}}{c_{m,n}}\right)-\sum_m \log\left(\frac{p_{m,k}}{c_{m,n}}\right)$
KL1	$\sum_m h_{k,n}\,c_{m,n}\log\left(\frac{h_{k,n}\,c_{m,n}}{p_{m,k}}\right)-h_{k,n}\,c_{m,n}+p_{m,k}$	$e^{-\sum_m c'_{m,n}\log\left(\frac{c_{m,n}}{p_{m,k}}\right)}$ with $c'_{m,n}=\frac{c_{m,n}}{\|\mathbf{c}_n\|_1}$	$1-e^{-\sum_m c'_{m,n}\log\left(\frac{c'_{m,n}}{p_{m,k}}\right)}$ with $c'_{m,n}=\frac{c_{m,n}}{\|\mathbf{c}_n\|_1}$
KL2	$\sum_m p_{m,k}\log\left(\frac{p_{m,k}}{h_{k,n}\,c_{m,n}}\right)-p_{m,k}+h_{k,n}\,c_{m,n}$	$\frac{1}{\sum_m c_{m,n}}$	$\sum_m p_{m,k}\log\left(\frac{p_{m,k}}{c'_{m,n}}\right)-p_{m,k}+c'_{m,n}$ with $c'_{m,n}=\frac{c_{m,n}}{\|\mathbf{c}_n\|_1}$

Table 3.1: Presentation of the measures of fit (the expressions assume $\|\mathbf{p}_k\|_1=1$)

The well-known **Euclidean distance** (*EUC*) defined by

$$D_{EUC}\left(\mathbf{x}|\mathbf{y}\right)=\sqrt{\sum_m \left(x_m-y_m\right)^2} \tag{3.5}$$

has already been used by Fujishima [1] for the chord recognition task.

The **Itakura-Saito divergence** [12] defined by

$$D_{IS}\left(\mathbf{x}|\mathbf{y}\right)=\sum_m \frac{x_m}{y_m}-\log\left(\frac{x_m}{y_m}\right)-1 \tag{3.6}$$

was presented as a measure of the goodness of fit between two spectra and became popular in the speech community during the seventies. This is not a distance : it is in particular not symmetrical. It can therefore be calculated in two ways : $D\left(h_{k,n}\,\mathbf{c}_n|\mathbf{p}_k\right)$ will define *IS1*, while $D\left(\mathbf{p}_k|h_{k,n}\,\mathbf{c}_n\right)$ will define *IS2*.

The **Kullback-Leibler divergence** [13] measures the dissimilarity between two probability distributions. It has been widely used in particular in information theory and has given rise to many variants : in the present paper we use the generalized Kullback-Leibler divergence defined by

$$D_{KL}\left(\mathbf{x}|\mathbf{y}\right)=\sum_m x_m\log\left(\frac{x_m}{y_m}\right)-x_m+y_m. \tag{3.7}$$

Just like Itakura-Saito divergence, the generalized Kullback-Leibler divergence is not symmetrical, so that we can introduce two measures of fit : $D\left(h_{k,n}\,\mathbf{c}_n|\mathbf{p}_k\right)$ *(KL1)* and $D\left(\mathbf{p}_k|h_{k,n}\,\mathbf{c}_n\right)$ *(KL2)*.

3.4 Filtering methods

So far our chord detection is done frame by frame without taking into account the results on the adjacent frames. In practice it is rather unlikely for a chord to last only one frame.

Furthermore the information contained in the adjacent frames can help decision : it is one of the main advantages of the methods using HMM, where the introduction of transition probabilities naturally leads to a smoothing effect. The post processing filtering we introduce works upstream on the calculated measures and not on the sequence of detected chords.

We introduce new criteria $\tilde{d}_{k,n}$ based on L successive values centered on frame n (L is then odd). These $\tilde{d}_{k,n}$ are calculated from the $d_{k,n}$ previously calculated on the L adjacent frames, as shown below. In our system two types of filtering are tested.

The **low pass filtering** defined by

$$\tilde{d}_{k,n} = \frac{1}{L} \sum_{n'=n-\frac{L-1}{2}}^{n+\frac{L-1}{2}} d_{k,n'} \tag{3.8}$$

tends to smooth the output chord sequence and to reflect the long-term trend in the chord change.

The **median filtering** defined by

$$\tilde{d}_{k,n} = \operatorname{med} \{d_{k,n'}\}_{n-\frac{L-1}{2} \leq n' \leq n+\frac{L-1}{2}} \tag{3.9}$$

has been widely used in image processing and is particularly efficient to correct random errors.

In every case, the detected chord $\hat{k}_n$ on frame n is the one that minimizes the set of values $\left\{\tilde{d}_{k,n}\right\}_k$:

$$\hat{k}_n = \underset{k}{\operatorname{argmin}}\ \tilde{d}_{k,n} \tag{3.10}$$

4 Evaluation and corpus

4.1 Beatles corpus

Our first evaluation database is made of the 13 Beatles albums (180 songs, PCM 44100 Hz, 16 bits, mono). This database is in particular the one used in MIREX 08 for the Audio Chord Detection task[1]. The evaluation is realized thanks to the chord annotations of the 13 Beatles albums kindly provided by Harte and Sandler [14]. In these annotation files, 17 types of chords are present (maj, dim, aug, maj7, 7, dim7, hdim7, maj6, 9, maj9, sus4, sus2, min, min7, minmaj7, min6, min9) and one 'no chord' label (N) corresponding to silences or untuned material. The alignment between annotations and wave files are done with the algorithm provided by Christopher Harte.

Figure 4.1 represents the repartition of the durations of the different chord types on the Beatles corpus. The most common chord types in the corpus are major, minor, dominant seventh, 'no chord' states, minor seventh and ninth. Any other chord type represents less than 1% of the total duration.

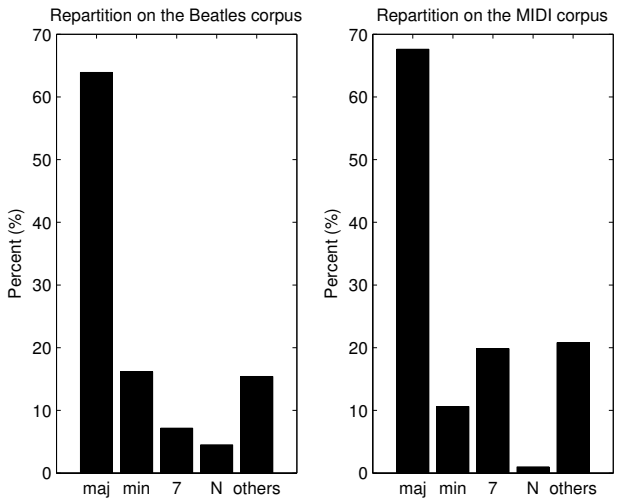

Figure 4.1: Statistics on the Beatles and the MIDI corpus : repartition of the chord types as percentage of the total duration.

[1]MIREX 08 (Music Information Retrieval Evaluation eXchange) : http://www.music-ir.org/mirex/2008/

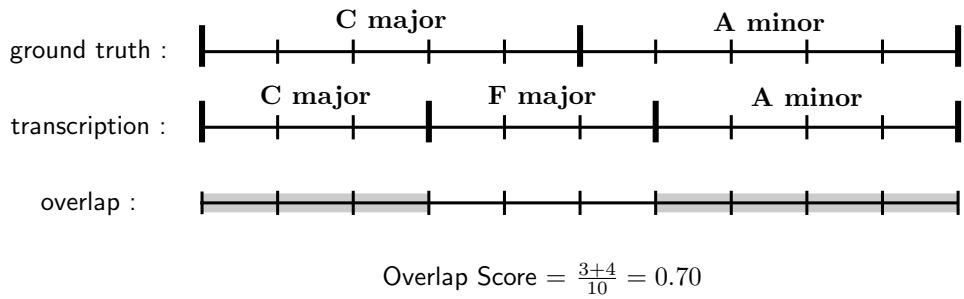

Figure 4.2: Example of calculation of an Overlap Score.

4.2 MIDI corpus

Our second evaluation database is composed of 12 songs from various artists in different genres (blues, country, pop and rock). The audio files (PCM 44100 Hz, 16 bits, mono) are synthesized from MIDI files[2] using the free software Timidity ++.[3] Timidity ++ is a software synthesizer which can generate realistic audio data from MIDI files using a sample-based synthesis method. We have manually annotated the songs : 5 types of chords are present (maj, min, 7, sus2, sus4) as well as the 'no chord' label (N). The repartition of the durations of the different chord types on the MIDI corpus is displayed on Figure 4.1.

4.3 Evaluation method

The evaluation method used in this paper corresponds to the one used in MIREX 08 for the Audio Chord Detection task.

This evaluation protocol only takes into account major and minor chord types. The 17 types of chords present in the annotation files are therefore first mapped into major and minor types following these rules :

- major : maj, dim, aug, maj7, 7, dim7, hdim7, maj6, 9, maj9, sus4, sus2
- minor : min, min7, minmaj7, min6, min9

For the systems detecting more chord types (dominant seventh, diminished, etc.), once the chords have been detected with their appropriate models, they are then mapped to the major and minor following the same rules than for the annotation files.

An *Overlap Score (OS)* is calculated for each song as the ratio between the lengths of the correctly analyzed chords and the total length of the song. We define for the Beatles corpus an *Average Overlap Score (AOS)* which is obtained by averaging the Overlap Scores of all the 180 songs of the corpus. An example of calculation of an Overlap Score is presented on Figure 4.2.

[2]The MIDI files were obtained on http://www.mididb.com
[3]The software is freely downloadable on http://timidity.sourceforge.net

4.4 Input features

Based on preliminary experiments we chose among three types of chromagram [7], [15], [16], the one proposed by Bello & Pickens [7], which appeared to give the best results for our chord transcription task. The Constant-Q Transform [2] allowing a frequency analysis on bins centered on logarithmically spaced frequencies is used. The center frequency f_k of the k^{th} bin is indeed defined as :

$$f_k = 2^{\frac{k}{b}} f_{min}, \tag{4.1}$$

where b represents the number of bins per octave and f_{min} the frequency where the analysis starts.

The signal is first downsampled to 5512.5 Hz and the CQ-Transform is calculated with $b = 36$ (3 bins per semi-tone), between frequencies 73.42 Hz (D2) and 587.36 Hz (D5). These parameters lead to a window length of 4096 samples and the hop size is set to 512 samples.

Thanks to the 36 bins per octave resolution, a tuning algorithm [3] can be used. After a pick detection in the chromagram, a correction factor is calculated so as to take into account the detuning. A median filtering is finally applied in order to eliminate too sharp transitions.

Some precisions about the calculation of the chromagram can be found in [7]. We used the code kindly provided by the authors. The silence ('no chord') detection is done by an empirically set threshold on the energy of the chroma vectors.

An example of chromagram and chord transcription is displayed on Figure 4.3.

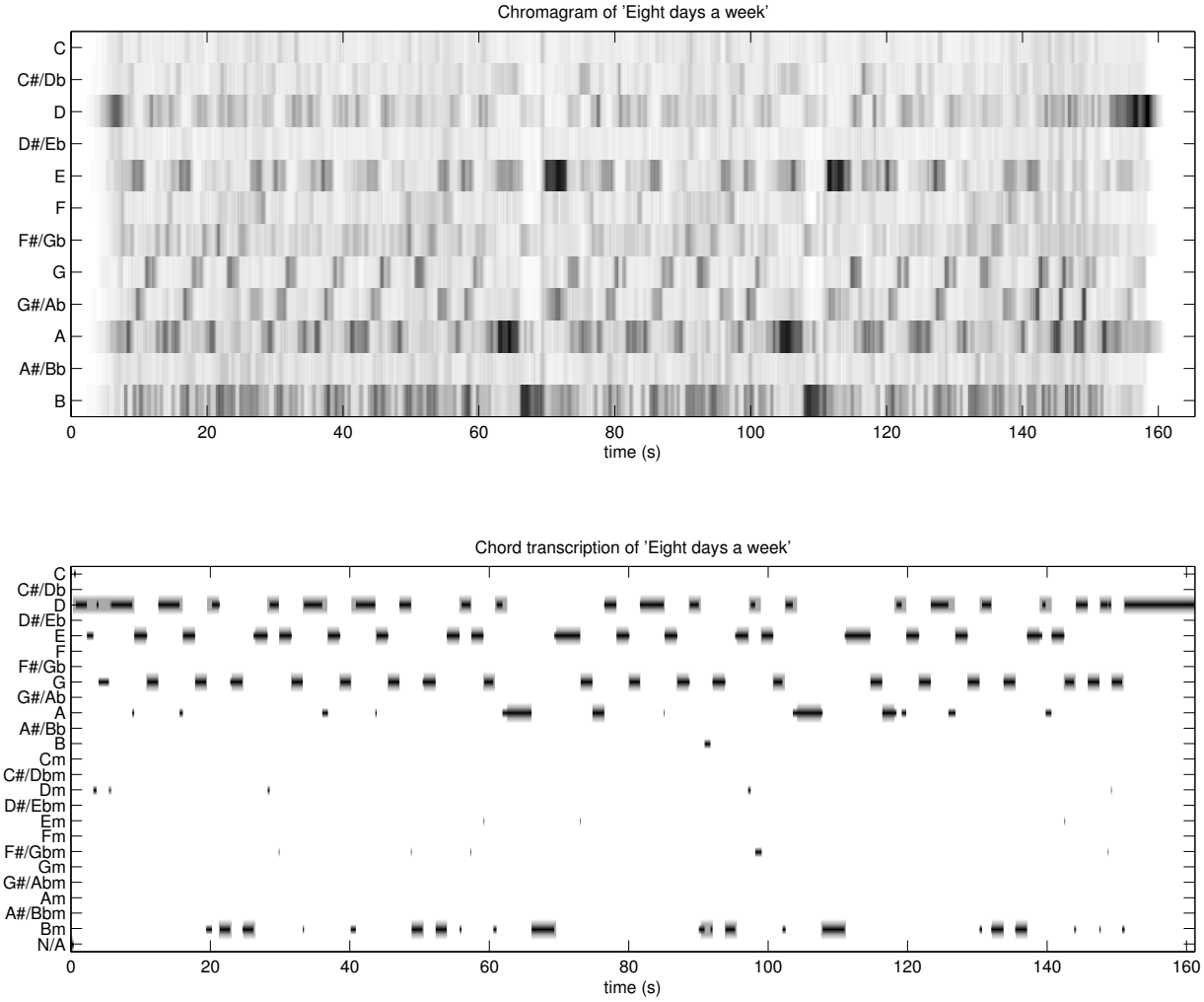

Figure 4.3: Chromagram and chord transcription for the song *Eight days a week* by The Beatles. At the bottom the estimated chord labels are in black while the ground-truth chord annotation is in gray.

5 Results on the Beatles corpus

The previously described five measures of fit (*EUC*, *IS1*, *IS2*, *KL1* and *KL2*), three chord models (1, 4 or 6 harmonics) and two filtering methods (low-pass and median) with neighborhood sizes from $L = 1$ to $L = 25$ are tested. These 375 parameter sets can be seen as so many chord recognition systems. We now investigate the systems giving the best results on the Beatles corpus.

5.1 Results with major/minor chord types

	no filtering			low-pass filtering			median filtering		
	1 harm.	4 harm.	6 harm.	1 harm.	4 harm.	6 harm.	1 harm.	4 harm.	6 harm.
EUC	0.665	0.636	0.588	0.710	0.684	0.646	0.705	0.679	0.636
IS1	0.665	0.441	0.399	0.706	0.460	0.415	0.706	0.465	0.422
IS2	0.657	0.667	0.170	0.704	0.713	0.178	0.703	0.714	0.178
KL1	0.665	0.487	0.140	0.700	0.532	0.151	0.692	0.498	0.143
KL2	0.667	0.672	0.612	0.709	0.712	0.648	0.714	**0.718**	0.656

Table 5.1: Average Overlap Scores on the 13 Beatles albums

Average Overlap Scores on the 13 Beatles albums are presented on Table 5.1 with the major/minor templates. For sake of conciseness, we only displayed the results for the optimal choice of L. The best average result is obtained with the KL2, the 4 harmonics chord model and the median filtering with $L = 15$ giving a recognition rate of 71.8%.

Interestingly, we notice two trends in the influence of the number of harmonics considered. We observe that for the EUC, IS1 and KL1, the results worsen when we increase the number of harmonics. In the particular cases of IS1 and KL1, this can be explained by the fact that they both contain a logarithm component which is sensitive to the zeros in the chord templates for the chord discrimination. We believe that since a high number of harmonics leads to a chord model with a low number of null components (see Figure 3.1), the discrimination between chords is harder, which leads to worse results. On the contrary, we observe that for the IS2 and the KL2, the best results are obtained by considering 4 harmonics.

A pathological situation appears with the Itakura-Saito divergences IS1 and IS2 with the 6 harmonics chord model. Indeed, we observe that the use of IS2 with the 6 harmonics chord model leads to a systematic detection of minor chords, while the IS1 measure with 6 harmonics chord model only detects major chords. In the case of the IS1 the loss in the scores

is less noticeable, because of the high number of major chords in the Beatles corpus. We believe that the explanation of this phenomena lies in the structure of the 6 harmonics chord model. Indeed, the 6 harmonics chord model gives a different number of null components for the major and minor chords : we can see on Figure 3.1 that the major chord model has 6 null components while the minor chord has 5 null components. The minimization criterion associated to the IS2 has the property that given a chroma vector, the more zeros in the chord template $\mathbf{p}_k$, the larger the value of the criteria. This measure of fit will therefore always give larger values for the chord models having more null components, that is to say the major chords, which leads to a systematic detection of only minor chords. The same phenomenon can be observed for the IS1 measure of fit, this time with a systematic detection of major chords.

Filtering clearly enhances the results : this can be explained by the natural proportion of chords to last more than the length of one frame. Both low-pass filtering and median filtering give good results : the low-pass filtering tends to smooth the chord sequence while the median filtering reduces the random errors. In most cases the optimal value of L is between 13 and 19 which corresponds, with our window parameters, to a length of approximately 2 seconds.

Some songs give bad results (<0.100) with all sets of parameters : it is often due either to a very strong detuning of the instruments which is too large to be corrected by the tuning algorithm present in the chromagram computation *(eg. Wild Honey Pie, Lovely Rita)*, or to un-tuned material such as spoken voice, non-harmonic instruments or experimental noises (applause, screams, car noise, etc.) *(eg. Revolution 9)*.

5.2 Results with other chord types

The simplicity of our method allows to easily introduce chord templates for chord types other than major and minor : we study here the influence of the chord types considered over the performances of our system.

Chord types	AOS
maj - min	0.718
maj - min - 7	0.724
maj - min - 7 - min7	0.706
maj - min - 7 - min7 - 9	0.706

Table 5.2: Results of the introduction of new chord types

The introduction of models for new chords types are tested on Table 5.2. The choice of the introduced chord types is guided by the statistics on the corpus previously presented. We introduce in priority the most present chords of the corpus : dominant seventh *(7)*, minor seventh *(min7)* and ninth *(9)*. For every method we only present the results for the optimal parameters (measure of fit, chord models, filtering method and neighborhood size).

The best results are obtained by detecting major, minor and dominant seventh chords, with the KL2, the one harmonic chord model and the median filtering with $L = 17$ giving a recognition rate of 72.4%.

The introduction of dominant seventh chords, which are very present in the Beatles corpus, clearly enhances the results. Yet, the introduction of more chord types which are less present

(minor seventh, ninth) degrades the results. Indeed, the introduction of a model for a new chord type gives a better detection for chords of this type but also leads to new errors such as false detections. Therefore only frequent chord types should be introduced, ensuring that the enhancement caused by the better recognition of these chord types is larger than the degradation of the results caused by the false detections.

5.3 Comparison with the state-of-the-art

Our method is now compared to the following methods that entered MIREX 08.

Bello & Pickens [7] use 24-states HMM with musically inspired initializations, Gaussian observation probability distributions and EM-training for the initial state distribution and the state transition matrix.

Ryynänen & Klapuri [8] use 24-states HMM with observation probability distributions computed by comparing low and high-register profiles with some trained chord profiles. EM-training is used for the initial state distribution and the state transition matrix.

Khadkevich & Omologo [17] use 24 HMMs : one for every chord. The observation probability distributions are Gaussian mixtures and all the parameters are trained through EM.

Pauwels, Verewyck & Martens [18] use a probabilistic framework derived from Lerdahl's tonal distance metric for the joint tasks of chords and key recognition.

These methods have been tested with their original implementations on the same Beatles corpus than before and evaluated with the same protocol (AOS). Results of this comparison with the state-of-the-art are presented on Table 5.3.

	AOS	Time
Our method (Maj-Min-7)	0.724	796s
Our method (Maj-Min)	0.718	790s
Bello & Pickens	0.707	1619s
Ryynänen & Klapuri	0.705	1080s
Khadkevich & Omologo	0.663	1668s
Pauwels, Varewyck & Martens	0.647	12402s

Table 5.3: Comparison with the state-of-the-art

First of all it is noticeable that all the methods give rather close results : there is only a 8% difference between the methods giving the best and worse results. Our methods give the best results, but more importantly with a very low computational time. There are indeed twice as fast as the best state-of-the-art method (Bello and Pickens).

5.4 Analysis of the errors

In most chord transcription systems, the errors are often caused by the structural similarity (common notes) and the harmonic proximity between the real chord and the wrongly detected chord.

Two chords are likely to be mistaken one for another when they *look alike*, that is to say, when they share notes (especially in template-based systems). Given a major or minor chord,

there are 3 chords which have 2 notes in common with this chord : the parallel minor/major, the relative minor/major (or submediant) and the mediant chord.

Besides the structural similarity, errors can also be caused by the harmonic proximity between the original and the detected chord. Figure 5.1 pictures the doubly nested circle of fifths which represents the major chords (capital letters), the minor chords (lower-case letters) and their harmonic relationships. The distance linking two chords on this doubly nested circle of fifths is an indication of their harmonic proximity.

Given a major or minor chord, the 4 closest chords on this circle are the relative (submediant), mediant, subdominant and dominant. One can notice that these 4 chords are also structurally close to the original chord, since they share 1 or 2 notes with it.

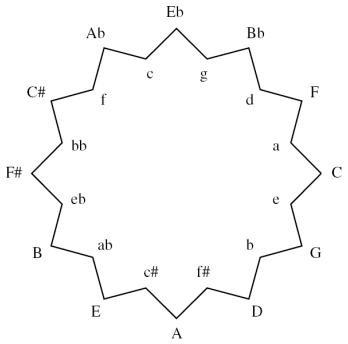

Figure 5.1: Doubly nested circle of fifths [7].

We have therefore brought out 5 potential sources of errors among the 23 possible ones (i.e., the 23 other wrong candidates for one reference chord). Examples of these potential sources of errors for C major and C minor chords are displayed on Table 5.4.

Reference chord	**C**	**Cm**
parallel	Cm	C
relative (submediant)	Am	A♭
mediant	Em	E♭
subdominant	F	Fm
dominant	G	Gm

Table 5.4: Particular relationships between chords and potential sources of errors : examples for C major and C minor chords

Figure 5.2 displays the repartition of these error types as a percentage of the total number of errors for every evaluated method. Errors due to the bad detection of the 'no chord' states are represented with the 'no chord' label.

The main sources of errors correspond to the situations previously described and to the errors caused by silences ('no chord'). Actually, in most methods, the 5 types of errors previously considered (over the 23 possible ones) represent approximately 60% of the errors.

The introduction of the dominant seventh chords clearly reduces the proportion of the errors due to relative (submediant) and mediant (-11%). Another noteworthy result is that the methods by Ryynänen & Klapuri, Bello & Pickens and our major/minor method approximately have the same error repartition despite the different structures of the methods, which proves that the semantic of the errors is inherent to the task. Pauwels, Varewyck &

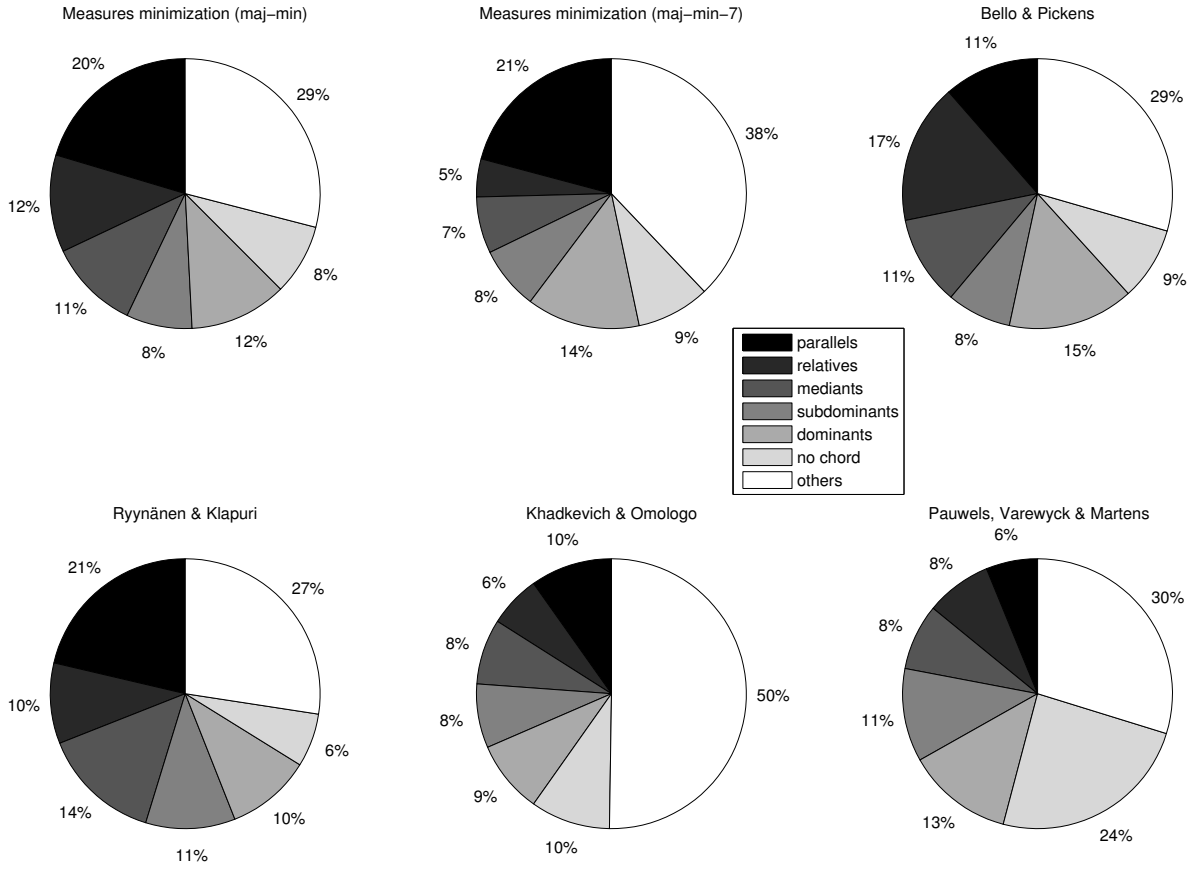

Figure 5.2: Repartition of the errors as a percentage of the total number of errors.

Martens' system is mostly penalized by the wrong detection of the 'no chord' states, when Khadkevich & Omologo's method produces a wider range of errors.

6 Results on the MIDI corpus

We now use the two sets of parameters described in the previous section for the Maj-Min *(KL2, 4 harmonics, median filtering with $L = 15$)* and the Maj-Min-7 *(KL2, 1 harmonic, median filtering with $L = 17$)* chord detection systems on the MIDI corpus.

6.1 Influence of the music genre

		Maj-Min		Maj-Min-7	
	Song title	Default	Optimal	Default	Optimal
Country	Ring of fire *(Johnny Cash)*	0.844	0.918	0.848	0.924
	Tennessee waltz *(Roy Acuff)*	0.941	0.955	0.949	0.955
	Stand by your man *(Tammy Wynette)*	0.895	0.909	0.902	0.911
Pop	Dancing queen *(ABBA)*	0.786	0.804	0.728	0.782
	I drove all night *(Cyndi Lauper)*	0.870	0.891	0.856	0.889
	Born to make you happy *(Britney Spears)*	0.867	0.892	0.861	0.892
Blues	Blues stay away from me *(The Delmore Brothers)*	0.630	0.791	0.854	0.912
	Boom, boom, boom *(John Lee Hooker)*	0.839	0.903	0.876	0.913
	Keep it to yourself *(Sonny Boy Williamson)*	0.771	0.909	0.907	0.928
Rock	Twist and shout *(The Beatles)*	0.827	0.892	0.850	0.901
	Let it be *(The Beatles)*	0.835	0.876	0.876	0.880
	Help ! *(The Beatles)*	0.918	0.920	0.899	0.918

Table 6.1: Overlap Score for the 12 songs of the MIDI corpus

Table 6.1 shows the Overlap Scores for the 12 songs of the MIDI corpus for the Maj-Min and the Maj-Min-7 chord recognition methods. Besides the results obtained with the default parameters, we also displayed the results with the optimal parameters in order to evaluate the fitness of our default parameters.

The first thing we can observe is that the scores obtained with the default parameters are rather close to the best ones. This shows that the parameters we deduced from the Beatles corpus can be used in a more general context.

We can also see that the scores are all creditable. This can surely be explained by the fact that we work here with resynthesized wave files and not real audio. These audio files are indeed generated with instrument patterns which contain less noise and untuned material than real instrument recordings.

Genre does not seem to have an influence on the scores. Nevertheless, the scores obtained on country songs are particularly large, but it is probably due to the very simple chord structures of these songs (mainly alternation of 3 chords).

6.2 Influence of the percussive noise

Our method strongly relies on the chromagram, that is to say a harmonic representation of the music. It can therefore be thought that inharmonic components of the music, for example drums, tend to add noise to the chromagram, which can lead to errors in the chord detection.

Working with audio data computed from MIDI files gives us the chance to synthesize them without the percussive parts. Indeed, the software Timidity ++ allows to mute one channel (instrument) for the wave-synthesis of the MIDI file.

The same simulations have been performed with these drum-free audio files. The removal of the percussions does not improve significantly the Overlap Scores. Indeed, the average score improvement is only 0.8% as well with the Maj-Min system than with the Maj-Min-7. We believe that the noise contained in the chromagram, which lead to errors, is not only due to drums but also, for example, to the melody itself, since it does not only play notes contained in the chord pattern.

6.3 Beat synchronous chord detection

The filtering process we have been using so far has a fixed length predetermined by the system parameters. It seems interesting to introduce beat information either in the chromagram computation or in the recognition criteria. For our tests we used the beat-detection algorithm provided by Davies & Plumbley [19].

The first way to take into account the beat information is to compute a beat-synchronous chromagram, that is to say averaging the chromagram over the number of frames representing a beat time. This process has already been used by Bello & Pickens [7]. Yet this does not improve the results : comparing the best results obtained with the usual chromagram and those obtained with the beat-synchronous one, it appears that the average degradation is -6% for the Maj-Min and -7% for the Maj-Min-7 system.

The second way to integrate this information is to filter the recognition criteria (either with the low-pass or the median filtering method) with a neighborhood size equal to the beat time. Even if the degradation is lower than with the beat-synchronous chromagram, the results are also penalized : the average degradation is -2% for the Maj-Min and the -4% for the Maj-Min-7 system.

We believe that these disappointing results are probably due to the fact that the beat detection does not take into account the distinction between on-beats and off-beats. Indeed, the chord change tend to occur mainly on the on-beats and not on every beat. Averaging either the chromagram or the recognition criteria on every beat does not really capture the rhythmic information.

7 Conclusion

In this paper we have presented a fast and efficient chord recognition method. The main innovative idea is the joint use of popular measures which had never been considered for this task and filtering methods taking advantage of time persistence. The decoupling of various stages of the chord template matching process enables to achieve high effectiveness in less time. Our system also offers a novel perspective about chord detection, which distinguishes from the predominant HMM-based approaches.

Since our method is only based on the chromagram no information about style, rhythm or instruments is needed so that our recognition system would work with any type of music. Furthermore we do not require any training on any database, which enables the computation time to be kept really low.

Acknowledgment

The authors would like to thank J. Bello, M. Khadkevich, J. Pauwels, M. Ryynänen and M. Davies for making their code available. We also wish to thank C. Harte for his very useful annotation files.

Bibliography

[1] T. Fujishima, "Realtime chord recognition of musical sound: a system using Common Lisp Music," in *Proceedings of the International Computer Music Conference (ICMC)*, Beijing, China, 1999, pp. 464–467.

[2] J. Brown, "Calculation of a constant Q spectral transform," *Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991.

[3] C. Harte and M. Sandler, "Automatic chord identification using a quantised chromagram," in *Proceedings of the Audio Engineering Society*, Barcelona, Spain, 2005.

[4] E. Gómez, "Tonal description of polyphonic audio for music content processing," in *Proceedings of the INFORMS Computing Society Conference*, vol. 18, no. 3, Annapolis, MD, 2006, pp. 294–304.

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

[6] A. Sheh and D. Ellis, "Chord segmentation and recognition using EM-trained hidden Markov models," in *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Baltimore, MD, 2003, pp. 185–191.

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

[8] M. Ryynänen and A. Klapuri, "Automatic transcription of melody, bass line, and chords in polyphonic music," *Computer Music Journal*, vol. 32, no. 3, pp. 72–86, 2008.

[9] K. Lee and M. 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*, vol. 16, no. 2, pp. 291–301, 2008.

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

[11] B. Pardo and W. Birmingham, "Algorithms for chordal analysis," *Computer Music Journal*, vol. 26, no. 2, pp. 27–49, 2002.

[12] F. Itakura and S. Saito, "Analysis synthesis telephony based on the maximum likelihood method," in *Proceedings of the International Congress on Acoustics*, Tokyo, Japan, 1968, pp. 17–20.

[13] S. Kullback and R. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[14] C. Harte, M. Sandler, S. Abdallah, and E. Gomez, "Symbolic representation of musical chords: A proposed syntax for text annotations," in *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, London, UK, 2005, pp. 66–71.

[15] G. Peeters, "Musical key estimation of audio signal based on hidden Markov modeling of chroma vectors," in *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, Montreal, Canada, 2006, pp. 127–131.

[16] Y. Zhu, M. Kankanhalli, and S. Gao, "Music key detection for musical audio," in *Proceedings of the International Conference on Multimedia Modeling (MMM)*, Melbourne, Australia, 2005, pp. 30–37.

[17] M. Khadkevich and M. Omologo, "Mirex audio chord detection," Abstract of the Music Information Retrieval Evaluation Exchange, 2008.

[18] J. Pauwels, M. Varewyck, and J.-P. Martens, "Audio chord extraction using a probabilistic model," Abstract of the Music Information Retrieval Evaluation Exchange, 2008.

[19] M. Davies and M. Plumbley, "Context-dependent beat tracking of musical audio," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 3, pp. 1009–1020, 2007.

Dépôt légal : 2009 – 4ème trimestre
Imprimé à l'Ecole Nationale Supérieure des Télécommunications – Paris
ISSN 0751-1345 ENST D (Paris) (France 1983-9999)

TELECOM ParisTech

Institut TELECOM - membre de ParisTech

46, rue Barrault - 75634 Paris Cedex 13 - Tél. + 33 (0)1 45 81 77 77 - www.telecom-paristech.frfr

Département TSI