AUTOMATIC CHORD RECOGNITION FOR MUSIC CLASSIFICATION AND RETRIEVAL

Heng-Tze Cheng, Yi-Hsuan Yang, Yu-Ching Lin, I-Bin Liao, and Homer H. Chen*

Department of Electrical Engineering, National Taiwan University
*Multimedia Applications Lab, Telecommunication Laboratories, Chunghwa Telecom

ABSTRACT

As one of the most important mid-level features of music, chord contains rich information of harmonic structure that is useful for music information retrieval. In this paper, we present a chord recognition system based on the N-gram model. The system is time-efficient, and its accuracy is comparable to existing systems. We further propose a new method to construct chord features for music emotion classification and evaluate its performance on commercial song recordings. Experimental results demonstrate the advantage of using chord features for music classification and retrieval.

Index Terms— Chord, music classification, N-gram

1. INTRODUCTION

Due to the fast growth of digital music collection and media playback on portable devices, effective retrieval and management of music is needed in the digital era. Most existing work in music information retrieval analyzes music via low-level features such as Mel frequency cepstral coefficient (MFCC) and other spectral coefficients. However, low-level features are insufficient for many applications since they are related to the signal characteristics rather than the semantic content of music. On the contrary, mid-level features such as chord, rhythm, and instrumentation represent musical attributes [1] and contain rich information for music analysis.

Chord sequence, which describes harmonic progression and tonal structure of music, is one of the most important mid-level features of music. Since harmonic progression is strongly related to the perceived emotion, similar chord sequences can be observed in songs that are close in genre, emotion, etc. With chord sequence, songs that are similar in various aspects can be identified and retrieved more effectively. In this paper, we focus on chord recognition and its applications.

However, existing research related to chord mainly focuses on chord transcription. Despite the potential of chord for cover song identification [2] and music segmentation [1] has been studied, other applications of chord have received little attention. In this paper, we present a novel method for chord feature construction and explore the application of the resulting new chord features—longest common chord subsequence and the histogram statistics of chords—to music emotion classification. We show the usefulness of chord features for emotion prediction. To our best knowledge, this is the first attempt to music classification and retrieval using chord features extracted from song recordings.

Furthermore, to extend our methods to real-time retrieval and recommendation of similar songs, we propose a new approach to chord recognition based on the N-gram model and the hidden Markov model (HMM). The approach is consistent with musical theory [1], and is more time-efficient than existing chord recognition systems, with comparable accuracy. The simplicity and time-efficiency of our approach are desirable for practical applications. Figure 1 shows the overall scheme and our major contributions.

The paper is organized as follows. Section 2 reviews previous work on chord recognition. The details of the proposed chord recognition system and the chord features are elaborated in Sections 3 and 4. Section 5 shows the experimental results, and Section 6 concludes the paper.

2. PREVIOUS WORK

Recent research on chord recognition is mainly based on HMM [6]. The general procedure is described as follows. First, the input audio is divided into successive frames and transformed to frequency domain. Then, feature vectors for chord recognition are extracted. Since chord consists of harmony formed by multiple notes or pitches, the 12-dimensional pitch class profile (PCP) feature vector or chroma vector, which represents the intensity of 12 semitone pitch class, is mostly used [1], [3], [4].

A chord recognition system involves both training and testing phases. In the training phase, the hidden Markov model is applied, where each state represents a single chord, and chord progression is modeled as a series of transition among states. The core elements in HMM are state transition probability and observation distribution. The state transition probability represents the likelihood that a chord is followed by another chord, and the observation distribution describes how likely an observed PCP feature

This work is supported by a grant from the National Science Council of Taiwan under the contracts NSC 96-2219-E-002-003 and NSC 96-2752-E-002-006-PAE.

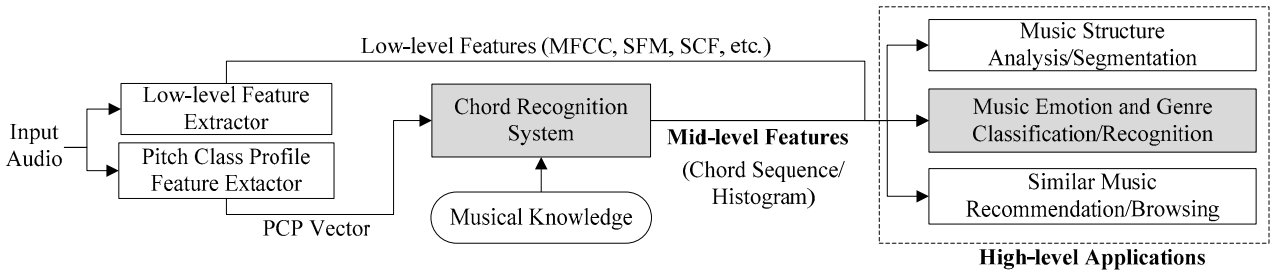

Figure 1. The scheme of achieving high-level applications using mid-level features. Gray-colored blocks are our major contributions.

vector is generated by a certain state. Different training methods have been used. In [3], HMM is trained via an expectation-maximization algorithm, and the observation distribution is modeled by a single Gaussian distribution. A similar approach is used in [1] except that musical knowledge is incorporated in the HMM initialization and that beat-synchronous segmentation is used to recognize chords according to beat times. In [5], the training data are extracted from MIDI files to train the HMM, and a unified model is presented to perform key estimation and chord transcription simultaneously.

In the testing phase, given the observed feature vectors and the pre-trained HMM, the most likely chord sequence is decoded using the Viterbi algorithm [1], [3]–[5]. Existing systems yield good accuracy around 70%. However, the Viterbi algorithm is time-consuming and the HMM for chord recognition is difficult to train. Due to the lack of massive training data and the varying nature of chord, over-fitting is inevitable.

3. CHORD RECOGNITION SYSTEM

In our system, we incorporate the N-gram model [6] to the HMM framework. In the training phase, the N-gram model is trained on ground-truth chord transcriptions to learn the common rules of chord progression. For each frame of the input audio in the testing phase, the chord with maximum likelihood is estimated using the pre-trained models. The main ideas of our approach are described in this section.

3.1. N-gram Model and Hidden Markov Model

Chord recognition can be effectively modeled using the basic concepts in digital speech processing. Inspired by the way human recognizes chords, we divide the task into two parts—acoustic modeling and language modeling:

- *Acoustic Modeling*: The most likely chord is estimated based on the observed PCP feature vector. A 24-state HMM is used to model 24 major/minor triad chords (e.g. C major, E minor, etc.). The goal is to find a chord that best fit the perceived music in a certain time interval.
- *Language Modeling*: Sequence of chord labels can be regarded as word sequence in natural language. Chord progression often follows the rules of harmony and appears in some common patterns (e.g. C→F→G→C), just like grammar and phrases in natural language. Hence, it is reasonable to train a language model that learns the common rule of chord progression.

Previous work mainly deals with acoustic modeling, leaving the issue of language modeling unaddressed. We adopt the N-gram model here for several reasons:

- In chord progression, a chord highly depends on its previous chords since successive chords generally share some common notes, which provide harmonic continuity to a music passage. The N-gram model is consistent with the property of chord progression in that the likelihood of an element in the sequence depends on the previous N elements. More formally, given a chord sequence $(c_1, c_2, \cdots, c_{i-1})$, the N-gram model predicts the next chord c_i based on the probability $P(c_i \mid c_{i-N}, c_{i-N+1}, \cdots, c_{i-1})$.
- Existing HMM-based systems consider only the transition probability between two consecutive chords. However, the information of two adjacent chords is insufficient for recognizing longer chord sequences.
- The scale of chord lexicon (only 24 major/minor chords in most work) is small compared with the word lexicon of natural language, so is the number of permutations. This makes the training of N-gram model manageable.

In our system, we adopt the method described in [4] for PCP feature vector extraction. Also, we recognize chords according to beat times detected by a beat tracking system, *BeatRoot* [9]. In the training phase, the N-gram model is started with $N=2$ (i.e. bigram) based on the circle of fifths [1]. Transitions between chords that sound more consonant with each other are initialized with higher probability. Then, the N-gram models with N equal to 2, 3, and 4 are trained on 152 hand-labeled chord transcriptions provided by Harte and Sandler [7]. For each song, we concatenate the labeled chords into text strings and train the N-gram model using *SRILM* [10], a toolkit for language modeling.

HMM training typically requires the information of chord boundaries. However, the precise time boundary between consecutive chords is often ambiguous in music and thus difficult to obtain, plus that there are very few databases available since labeling the chord boundaries is very laborious. Our approach is free of such problems because the N-gram model is trained on chord sequences only. Experimental results in Section 5 indeed show that our approach is general enough for a large variety of songs.

3.2. Chord Decoding

In the testing phase, the chord sequence is decoded from the input audio. Let $\mathbf{O}_i$ be the observed PCP feature vector and c_i the decoded chord of the ith frame. In the case where bigram and trigram models are used, the chord c_i^* with maximum likelihood is obtained by

$$c_i^* = \arg\max_{c_i} \left(\alpha \cdot \log P_{bi} + \beta \cdot \log P_{tri} + (1-\alpha-\beta) \cdot \log P(c_i \mid \mathbf{O}_i) \right) \quad (1)$$

where $P_{bi} = P(c_i \mid c_{i-1})$, $P_{tri} = P(c_i \mid c_{i-2}, c_{i-1})$ and α and β , respectively, are nonnegative weights given to P_{bi} and P_{tri} , and $\alpha + \beta \le 1$. Eq. (1) shows how we use HMM and N-gram model for acoustic and language modeling, respectively. $P(c_i \mid \mathbf{O}_i)$, the probability that $\mathbf{O}_i$ is generated by c_i , is estimated by calculating the correlation between $\mathbf{O}_i$ and a set of chord templates as described in [4]. P_{bi} and P_{tri} are given by the N-gram model.

The time complexity of our method is one order less than that of the Viterbi algorithm [6]. Consider a song with n frames long. Both methods involve calculating the observation probability and the chord transition probability. In the Viterbi algorithm, the optimal previous chord for each candidate chord is estimated for each frame, and the paths are stored in a table. Suppose there are k candidate chords in the HMM; then k^2 such operations are required for each frame, and the total time complexity is $O(k^2 n)$. On the contrary, only k such operations are needed in our method for each frame since we simply examine the k candidate chords to determine an optimal one. The total time complexity is $O(kn)$.

4. CHORD FEATURES AND APPLICATIONS

4.1. Longest Common Chord Subsequence

To measure the similarity between two chord sequences, we calculate the longest common chord subsequence (LCCS) via a dynamic programming algorithm [8] that has been used in bioinformatics for DNA sequence comparison.

Since the order of chords cannot be rearranged, LCCS as a similarity metric is desirable because it can capture the similarity between sequences while preserving their order. For example, given two chord sequences, s_1 = (C, F, G, C) and s_2 = (C, Am, F, Dm, G), the LCCS of them is s_{LCCS} = (C, F, G). We can normalize LCCS by the length of sequence to facilitate the comparison of different songs. Given any two songs, the longer the LCCS is, the more similar they are.

4.2. Chord Histogram

We propose another feature called chord histogram (CH) to show the percentage of time each chord occupies in a song. As illustrated in Figure 2, we can see that the chords C, F, G and Am frequently appear in both songs. Although these chords may appear in different orders, they can still arouse similar emotions. We can also get the sense of tonality from the chord histogram without going through a key detection process in advance. To measure the similarity between two songs, the L2 distance is used.

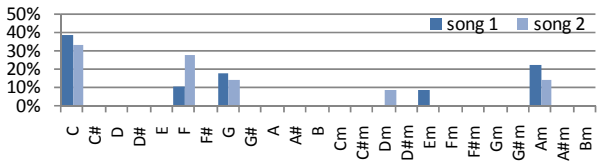

Figure 2. A chord histogram showing the chord similarity between two songs in our database.

4.3. Application on Music Emotion Classification

While the LCCS indicates the chord similarity in order, the chord histogram sketches an image of the harmonic structure of a song. These two proposed chord features are used for music emotion classification.

Recent research on music emotion classification models the emotion in terms of arousal (how exciting/calming) and valence (how positive/negative). While arousal can be easily predicted by loudness and tempo, there are few salient low-level features related to valence. The difficulty in predicting valence is pointed out in [9] and many other papers.

In order to solve the problem, we model the valence using chord features. Since major chords can generally arouse positive mood while minor chords are more negative, it is reasonable to infer that harmonic progression can affect valence. We focus our work on valence prediction because there is no obvious relationship between chords and arousal.

5. EXPERIMENTAL RESULTS

5.1. Evaluation on Chord Recognition System

To compare our system with existing work, we use the test set provided by Harte and Sandler [7], which contains 28 recordings from 2 albums, *Please Please Me* (CD1) and *Beatles for Sale* (CD2), of the Beatles. Evaluation is made on a frame-by-frame basis, and only the exact matches between the decoded chord sequences and the ground-truth transcriptions are counted as correct recognitions. Results are shown in Table I.

The best results are achieved using trigram and 4-gram model. The N-gram-based approach outperforms the typical HMM-based approach without N-gram by 7% in overall accuracy. This supports the claim that the N-gram model can effectively learn the common rules of chord progression.

The same test set is used in [5] and [7]. The system in [7] uses template matching in chord decoding and yields 62.4% overall accuracy, which is lower than ours (67.3%). An overall accuracy of 72.8% is achieved in [5] using key-independent HMM. However, our system is simpler and

more time-efficient as discussed in Section 3.

We can also see that the accuracy of CD1 increases significantly from 51.2% to 60.9% when 4-gram is used, while the accuracy of CD2 improves less, from 68.7% to 72.5%. Since more ambiguities are found to be present in the observed feature vectors of CD1 [5], we can infer that the N-gram model is very helpful when the acoustic features are ambiguous. This is important since the acoustic features of a chord often vary with music genres and instrumentation.

5.2. Music Emotion Classification Using Chord Features

For music emotion classification, we evaluate our approach on the database proposed in [11], which consists of 195 popular songs of various genres and artists from Western, Chinese, and Japanese albums.

Besides the proposed chord features, LCCS and CH, we also include low-level features such as Mel frequency cepstral coefficient (MFCC), spectral flatness measure (SFM), and spectral crest factor (SCF), all extracted by *Marsyas* [12], in the evaluation. We use the *k*-nearest neighbor algorithm to find *k* most similar songs for each song and then set the valence of the song to be the same as that of the majority of the *k* songs.

The results are shown in Figure 3. The overall accuracy of valence prediction using low-level features is around 57%. The accuracy increases to 61.03% when only the two chord features are used. This result shows the strength of chord features for music emotion classification. Furthermore, we can see from Figure 3(a) that the chord features are useful for predicting positive valence, and the low-level features are helpful for negative valence. Hence, by using an early-fusion to concatenate the chord features and the low-level feature vectors for classification, we can enhance the overall accuracy to 63.08%, as shown in Figure 3(b). Note that although both LCCS and CH are more useful than low-level features, better results are achieved using CH since LCCS is more subject to errors in chord recognition. The advantage of LCCS can become more salient as the accuracy of chord recognition increases in the future.

6. CONCLUSION

In this paper, we have investigated mid-level music feature construction and its applications to music classification and retrieval. With recognition accuracy as competitive as existing systems and simplicity and time-efficiency advantages, the proposed N-gram-based chord recognition system is particularly attractive for practical applications.

The two proposed new chord features, longest common chord subsequence and chord histogram, are useful for music analysis, management, and retrieval. With these two mid-level music features, we are able to achieve 6% improvement over existing approaches that use only low-level features for emotional valence prediction.

Table I
Accuracy and time complexity of chord recognition systems

System		CD1	CD2	Overall	Complexity
Without N-gram		51.2%	68.7%	60.7%	$O(kn)$
N-gram Based	$N=2$	56.3%	70.3%	63.9%	$O(kn)$
	$N=3$	60.6%	73.0%	**67.3%**	$O(kn)$
	$N=4$	60.9%	72.5%	**67.2%**	$O(kn)$
Template Matching [7]		53.9%	70.8%	62.4%	$O(kn)$
HMM / Viterbi [5]		61.0%	84.5%	72.8%	$O(k^2n)$

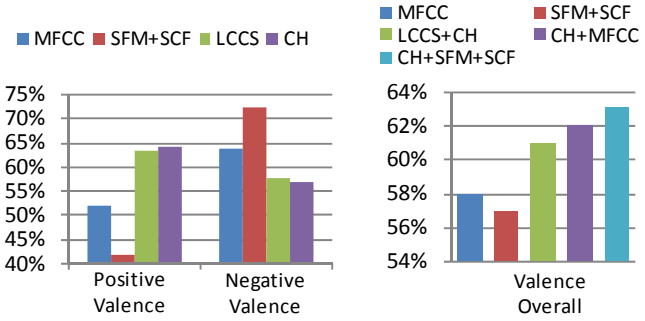

Figure 3. Accuracy of valence prediction using different feature sets. The incorporation of chord features (LCCS, CH) improves the overall accuracy to 63%.

7. REFERENCES

[1] J. P. Bello and J. Pickens, "A robust mid-level representation for harmonic content in music signals," in *Proc. ISMIR*, pp. 304–311, 2005.

[2] K. Lee, "Identifying cover songs from audio using harmonic representation," in *MIREX task on Audio Cover Song Identification,* 2006.

[3] A. Sheh and D. P. Ellis, "Chord segmentation and recognition using EM-trained hidden Markov models," in *Proc. ISMIR*, pp. 185–191, 2003.

[4] H. Papadopoulos and G. Peeters, "Large-scale study of chord estimation algorithms based on chroma representation," in *Proc. CBMI*, pp. 53-60, 2007.

[5] K. Lee and M. Slaney, "A unified system for chord transcription and key extraction from audio using hidden Markov models," in *Proc. ISMIR*, pp. 245-250, 2007.

[6] X. Huang, A. Acero, and H.-W. Hon, "*Spoken Language Processing: A Guide to Theory, Algorithm and System Development,"* Prentice Hall, 2001.

[7] C. Harte and M. Sandler., "Automatic chord identification using a quantised chromagram," *AES Convention*, 2005.

[8] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "*Introduction to Algorithms*," 2nd Ed., McGraw-Hill, 2001.

[9] S. Dixon, "Evaluation of the audio beat tracking system BeatRoot," *Journal of New Music Research*, pp. 39-50, 2007.

[10] A. Stolcke, "SRILM — An extensible language modeling toolkit," in *Proc. ICSLP*, 2002.

[11] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. H. Chen, "A regression approach to music emotion recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 2, pp. 448-457, Feb. 2008.

[12] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE TSAP*, vol. 10, pp. 293–302, 2002.