Cover Song Identification by Sequence Alignment Algorithms

Chih-Li Wang[1], Qian Zhong[2], Szu-Ying Wang[3] and Vwani Roychowdhury[4]
[1,2,4]Electrical Engineering, UCLA
[3]Department of Music, NHCUE
[1]chihliwang@ucla.edu,[2]qianz@ucla.edu , [3]p96051@mail.nhcue.edu.tw,[4]vwani@ee.ucla.edu

ABSTRACT

Content-based music analysis has drawn much attention due to the rapidly growing digital music market. This paper describes a method that can be used to effectively identify cover songs. A cover song is a song that preserves only the crucial melody of its reference song but different in some other acoustic properties. Hence, the beat/chroma-synchronous chromagram, which is insensitive to the variation of the timber or rhythm of songs but sensitive to the melody, is chosen. The key transposition is achieved by cyclically shifting the chromatic domain of the chromagram. By using the Hidden Markov Model (HMM) to obtain the time sequences of songs, the system is made even more robust. Similar structure or length between the cover songs and its reference are not necessary by the Smith-Waterman Alignment Algorithm.

Keywords-Cvoer song; Hidden Makrov Model; Smith-Waterman Algrortihm

1. INTRODUCTION

Content-based music analysis has raised extensive interest in recent years. The ever pervasive use of internet makes acquiring digital music increasingly easy. However, digital recordings are not always tagged with appropriate metadata especially for older recordings. Therefore it has become more challenging to organize songs into genres, to locate a specific song or to find a cover version of a song from thousands of songs.

Many researchers have focused on music modeling using the Mel Frequency Cepstral Coefficient (MFCC) [1], including measuring the music similarity [2] [3] [4], classifying the genres [5] [6] and identifying the artist [7] [8]. More recently Venkatachalam *et al* [9] [10] have achieved great success in identifying the exact song recordings from different sources.

Despite of the significant progress, cover song identification still remains challenging. A cover version of a song is a new production of an old composition. Typically, a cover song keeps the melody of the original song but may alter other critical properties, such as the performing artists, instruments, rhythm and structure. Thus the exact matching algorithm proposed in [9] [10] is not quite suitable for the cover song identification task. Moreover, a feature or algorithm independent of the timber, tempo or structure in a nutshell is needed for the cover song identification task. To this end, while widely used in music modeling [1-8], the MFCC is also inadequate for cover song identification due to its sensitivity to the timber.

Several other research groups [11] [12] [13] have used the pitch class profile (chromagram) [14] to extract the chords or keys, which demonstrated greater promise in this area. Chromagram captures the melody similarity and is less sensitive to the timber. Utilizing such feature, the best performance on the cover song identification task was previously achieved at the 2006 MIREX conference [15].

In this paper, a new method (Figure **1**) for cover song identification is proposed. Instead of using the cross-correlation of the beat-synchronous chromagram directly [15], we use the state sequence derived from chromagram's HMM, which further improves the model robustness. In particular, the beat/chroma-synchronous chromagram is firstly extracted for every query. The Hidden Markov Model (HMM) is then trained via the expectation maximization (EM) algorithm based on the song's beat/chroma-synchronous chromagram, followed by estimating the most probable state sequence via the Viterbi algorithm. After this step, the reference song's beat-synchronous chromagram is extracted and cyclically shifted in the chromatic domain in order to capture the key transposition. The most probable state sequences are thus estimated for all shifted reference song's chromagrams from all query songs' HMMs. Finally the Smith-Waterman sequence alignment algorithm is applied to all state sequence pairs to identify the reference song for the query.

International Conference on Graphic and Image Processing (ICGIP 2011), edited by Yi Xie, Yanjun Zheng,
Proc. of SPIE Vol. 8285, 82855M · © 2011 SPIE · CCC code: 0277-786X/11/$18 · doi: 10.1117/12.913429

2. METHOD

2.1 Chromagram Estimation

2.1.1Beat- synchronous chromagram

Cover songs and its reference song should have similar melody but may be performed by different artists, with different instrument or at different tempo. Therefore, a wrapped chromagram, which is less dependent on the timber but more sensitive to the melody, is used. It is derived by

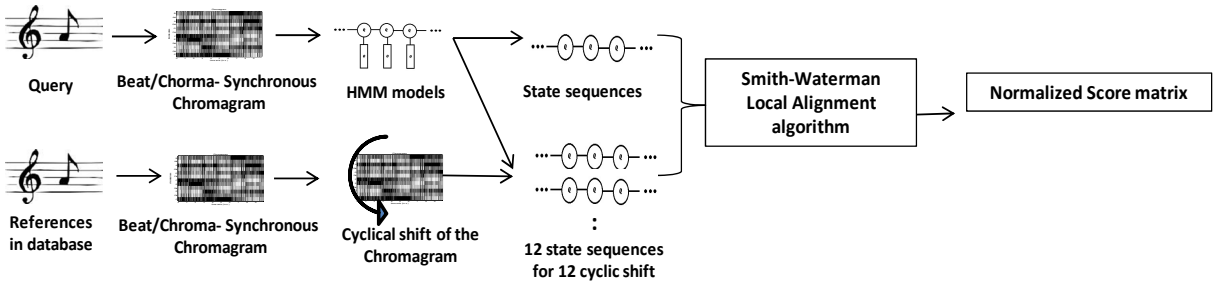

Figure 1. Method overview.

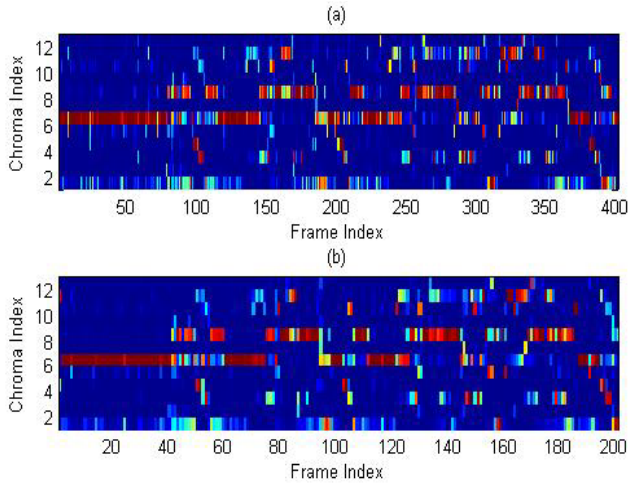

Figure 2. The chromagram of a section of all tomorrow's parties by Andy Warhol with Nico (a)The beat-synchronous chromagram (b) The chroma-synchronous chromagram. There are several large-valued chromatic classes in a chroma vector. Therefore, the components in a chroma vector are not independent. Also, the length of the beat-synchronous chromagram is double the length of the chroma-synchronous chromagram

mapping the frequencies onto the logarithm-spacing bins corresponding to the 12 chromatic scales (classes) over several octaves. Then, the chromas of different octaves are wrapped into one octave to form a 12-dimension wrapped chromagram.

Due to different tempo of the cover versions, the beat-synchronous chromagram --a chroma vector per beat-- is necessary and implemented by D Ellis and G Poliner [15]. Briefly, there are two steps to estimate the optimal beat time-: First, the onset strength is extracted by a log-magnitude 40 channels MFCC. Mel frequency, which maps the frequencies onto the mel-scale frequency bands, takes into account the human's perception of sounds and is more suitable for modeling of sounds than linear frequency. A MFCC is obtained by taking the discrete cosine transform (DCT) of the mel-frequency log-scale spectrum. The MFCC is sensitive to the timber of a sound, such as bass. Hence, the MFCC could be adopted to estimate the beat of a song. The global tempo is estimated by selecting the maximum autocorrelation of the onset strength at around 240BPM. Second, the optimal beat time is estimated by the dynamic programing. The

beat-synchronous chromagram is evaluated by averaging out the chroma vectors within a beat time. Hence, the beat-synchronous chromagram represents a chroma vector per beat time.

2.1.2Chroma-synchronous chromagram

The chromagram have better discriminability of the chromatic class than the MFCC. A note might be held for several beats. Therefore, we also try to utilize the wrapped chromagram instead of the MFCC to extract the onset strength and estimate the "chroma" time. The "chroma-synchronous chromagram" represents a chroma vector per chroma time. Figure 2 (a) and (b) show the beat-synchronous chromagram and the chroma-synchronous chromagram of "all tomorrow's parties" by Andy Warhol with Nico. We could observe that the length of the beat-synchronous chromagram is double the length of the chroma-synchronous chromagram

2.1.3Chroma-vector normaliztion

The cover versions may be performed with different volume levels. Hence, a chroma vector is raise to power 2 and then normalized to a unit norm.

$$Cn = \left(\frac{X^2}{\sum_{i=1}^{12}(x_i)^2}\right) \tag{1}$$

,where X is a chroma vector in the chromagram and $X = \{x_i\}$, $i = 1 \ldots 12$

2.2 Key transpostioin

Human's perception of pitch is less sensitive to the absolute pitch but the relative pitch. Therefore, the cover song and its reference could be performed by different key. Hence, a reference's chromagram is cyclically shifted in the chromatic domain to take the key transposition into consideration.

Figure 3 (a) and (c) are the chromagrams of the cover songs with different languages sung by Sun boy'z. The version in Figure 3 (c) is a note higher than in (a) and therefore if (a) is cyclically shifted 1 position up in the chromatic domain as shown in (b), the chromagrams of the two cover songs are highly similar.

2.3 Hidden Makove Models Estimation

A Hidden Markov Models (HMM) [16] comprise:

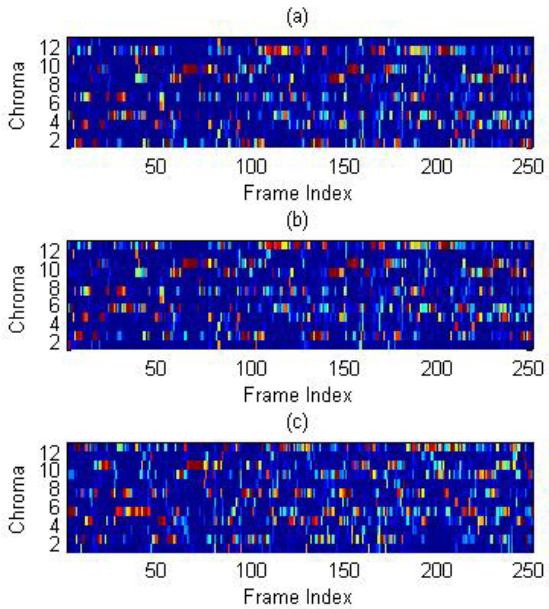

Figure 3. A pair of cover songs performed by a Hong Knog group, Sun boy'z. (a) is the chinese version, and (c) is the Cantonese version. The chinese version is one note lower than the Cantonese version. Therefore, if (a) is cyclicaly shifted 1 postion up in chromatic domain as shown in (b), the two chromagram, (b) and (c), are highly similar.

a) The observation sequence $O_1 \dots O_t$.

b) The hidden state sequence $Q_1 \dots Q_t$ *with* $\{Q\} \in S = \{S_1 \dots S_N\}$ *,where N is number of state.s*

c) The state transition probability:

$$A = \{a_{ij}\} = \{P(q_{t+1} = Sj | q_t = S_i)\}. \quad (2)$$

It obeys the Markov property - that is given the present, the future does not depend on the pass.

d) The emission probability (B). *Each state generates observations according to its emission probability.*

The parameters of the HMM model are normally denoted by $\lambda = (A, B, \pi)$. Given the observations, it could be derived by the Baum-Welch algorithm which is a special case of the Expectation-Maximization algorithm.

The E-step is to calculate the Baum's auxiliary function:

$$Q(\lambda, \bar{\lambda}) = \sum_Q P(Q|O, \lambda) log\left[P\left(O, Q\middle|\bar{\lambda}\right)\right] \quad (3)$$

The M-step is the maximization over $\bar{\lambda}$.

On the other hand, given the HMM model and the observations, the most likely state sequences could be obtained by the Viterbi algorithm.

In our method as shown in Figure 1, each query song has its HMM model. Each HMM model contains 5 hidden states. The chromagram of a song are used as the observed outputs of the HMM model. The emission probability is assumed a single Gaussian distribution with full covariance matrix since a chroma vector normally contains several large-valued chromatic classes as shown in Figure 2 and Figure 3; hence, the components in a chroma vector are not independent. The initial mean and the covariance of the emission probability for each state are estimated by the empirical mean and covariance of the k-mean clustering with k=5. The parameters of a HMM model are obtained via the Baum-Welch algorithm; finally, the most likely state sequence is solved by the Viterbi algorithm.

For a reference song in the database, after its beat/chroma-synchronous chromagram is extracted and cyclically shifted in the chromatic domain. The most probable state sequences are thus estimated for all shifted reference song's chromagrams from all query songs' HMMs.

2.4 State Sequence Alignment

The Smith-Waterman algorithm [17] is a dynamic programing algorithm and is used to determine the optimal local alignment of two sequences with respect to the scoring matrix (W) and gap penalty (G). A similarity matrix (S) for the two sequences is built by

$$S(i,j) = max\begin{cases} S(i-1, j-1) + W\left(State_{seq1}, State_{seq2}\right) \\ S(i-1, j-1) + G \end{cases} \quad (4)$$

, where $W\left(State_{seq1}, State_{seq2}\right)$ is the score between states of sequence 1 and sequence 2.

After getting S, the negative components in S are set to 0. The best local-aligned section is obtained by backtracking from the largest component in S until first 0 is encountered.

Since the reference song does not have the HMM models in our method, the empirical mean of the chroma vectors over time associated with a state is used to represent the feature of the state. Then, the pairwise Euclidian distances (P) between the states' features of two sequences are calculated. In order to make the score (W) of the aligned states between two sequences positive and the misaligned ones negative, the scoring matrix (W) is defined as

$$W = diag(1) - \frac{P}{max(P)} \quad (5)$$

Since the value of W are between ± 1, the negative gap penalties from 0 to 2.7 are evaluated.

The Smith-Waterman Alignment may perform better than the cross-correlation of the entire songs [15], especially when the structure of the cover version is different from the original one by reordering, inserting, deleting some parts of the original one. The beat estimation is not always accurate and inserting gaps could alleviate the estimation error. Our method is similar to J Bello [18]. However, prior knowledge about the musical chords in their method is needed due to the chord estimation. Knowing the beginning and the ending points are necessary when using the Needleman-Wunsh-Sellers algorithm (a global alignment algorithm) [18] or the dynamic time wrapping [19][20] such that these methods do not fit for the input query consisting only parts of the entire song.

3. RESULT

The performance is evaluated over 80 pairs of songs specified in [21]. For every pair, one song is randomly sampled as the query song, and the other one is reserved as the reference song. Hence there are 80 query songs in total. For every query song, there are 80 reference candidates in the database. Our goal is to identify the correct candidate for every query song.

As described in section 2, the chroma-synchronous chromagram and the beat-synchronous chromagram are firstly extracted from all reference songs and query songs. Then a HMM is trained for every query song's chromogram and its most likely state sequence is obtained by the Viterbi algorithm. Next each chromagram for the reference is cyclically shifted in the chromatic domain to capture the 12 key shifts. A query's HMM is then applied to the set of 80 reference songs' shifted chromogram to estimate the reference songs' state sequences. That's, $2 \times 12 \times 80$ reference songs' state sequences are estimated for every query song. Finally the Smith-Waterman sequence alignment algorithm is applied to all $2 \times 12 \times 80$ sequences to identify the correct reference for one query song.

Beat estimation is in general a difficult problem, and it might be inaccurate sometimes. In this case, the gap penalty in the sequence alignment plays a significant role in compensating for the incorrect estimation. However, when the gap penalty is too small, two extremely dissimilar sequences could be perfectly aligned by inserting many gaps. On the other hand, when the gap penalty is too large, the sequence alignment algorithm wouldn't use the gap to correct the beat-estimation error.

Figure 4 shows the average length of aligned sequence pairs for both correct and incorrect cover pairs at different gap penalties. It's observed that the incorrect cover pairs even assume longer aligned sequences than correct pairs when gap penalty approaches zero. And when the gap penalty is large than 1.20, the average length of the aligned sequences becomes saturated. In our experiment, a wide gap penalty range between 0.3 and 2.1 has been studied.

In addition, the alignment of longer sequences in general tends to generate a larger alignment score. If we record the raw alignment score between every query song and every reference candidate, we could construct a 80×80 raw score matrix. Following the steps described in Section 2, the raw alignment score matrix is normalized so that the sum of each row and column are equal to 1. Finally, a query will pair with a reference song with the highest normalized score.

Figure 5 shows the average normalized score for all correct and incorrect cover pairs at different gap penalty. As expected, the normalized score saturates when the gap penalty reaches 1.20.

Figure 6 shows the recall and precision for our method. The performance increases from 37 songs by paring only from the beat-synchronous method with gap penalty at 0.5 to 43 songs out of 80 songs by paring from both of the beat and chroma-synchronous methods. Also, when the score is above 0.035, the precision is about 85% in Figure 6. Furthermore, 26 songs out of the correct 43 pairs are paired by using the beat-synchronous chromagram method and the reminding are paired by the chroma-synchronous method. Hence, the chroma-synchronous chromagram method could improve the performance.

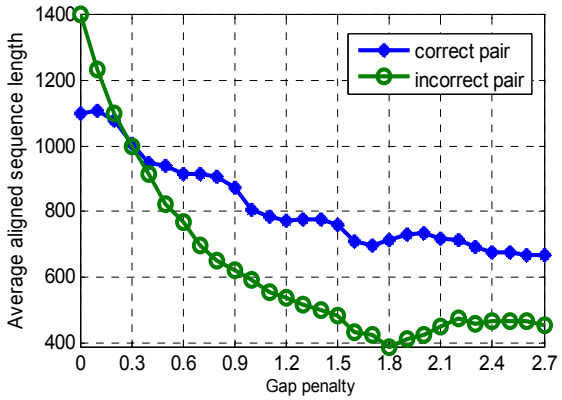

Figure 4. The gap penalty vs. the average length of the aligned segence pairs for the beat-synchronous chrmagram method. The incorrect cover pairs have longer aligned sequences than correct pairs when the gap penalty approaches zero since two extremely dissimilar sequences could be perfectly aligned by inserting many gaps. When the gap penalty is large than 1.20, the average length of the aligned sequences becomes saturated.

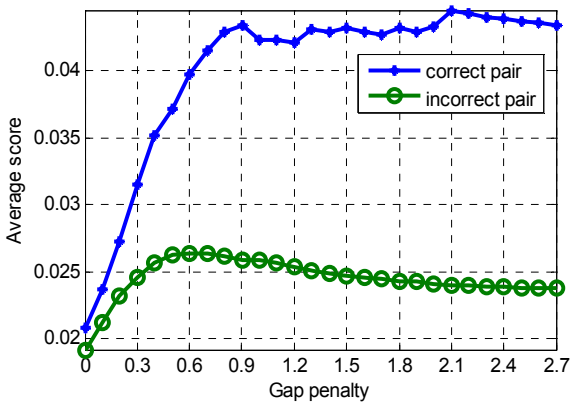

Figure 5. The average normalized score vs.the gap peanlty of the beat-synchronous chrmagram method. The normalized score saturates when gap penalty reaches 1.20.

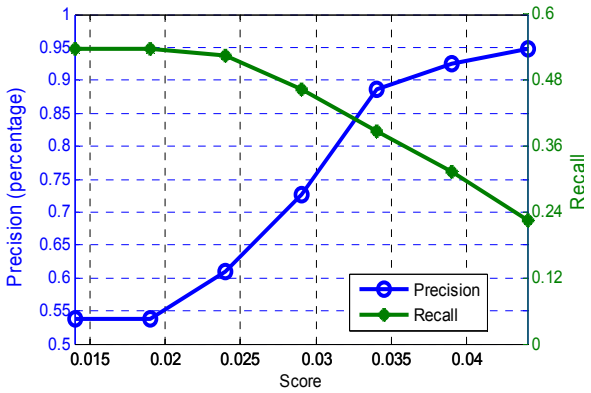

Figure 6. The precision and recall of the correctly paired songs above the indicated score vs. the score. When the score is above 0.035, the precision is about 85%

We applied D Ellis's and G Poliner's algorithm[1] [15] to this dataset [21] as well, and their algorithm correctly paired 31 songs. By using the HMM and the Smith-Waterman algorithm, we improved the system's performance by 15%.

4. FURTURE WORK

Future work includes evaluating our method in a large scale via the Music Information Retrieval Evaluation eXchange (MIREX), which is an organization to evaluate the system's performance from different groups. Moreover, our method might also be used to analyze the structure of a song by the sub-optimal self-alignment. We have tested that our method is good to find the repeating parts within a song. However, we still need to develop a more sophisticated method to eliminate short repeating parts, overlap sections and to detect the whole structure of a song after finding the replications.

REFERENCES

[1] B Logan, "Mel frequency cepstral coefficients for music modeling," *Proc. Int. Symp. on Music Information Retrieval*, Jan 2000.
J Aucouturier and F Pachet, "Music similarity measures: What's the use," *Proceedings of the ISMIR*, Jan 2002.

[2] A Berenzweig, B Logan, and D Ellis, "A large-scale evaluation of acoustic and subjective music-similarity measures," *Computer Music Journal*, Jan 2004.

[1] MATLAB code can be downloaded from http://labrosa.ee.columbia.edu/projects/coversongs/

[3] J Foote, "Visualizing music and audio using self-similarity," *Proceedings of the seventh ACM international conference on Multimedia*, Jan 1999.

[4] M Mandel and D Ellis, "Song-level features and support vector machines for music classification," *Proc. ISMIR*, Jan 2005.

[5] D Pye, "Content-based methods for the management of digital music," *IEEE International Conference on* Acoustics *Speech and Signal Processing*, Jan 2000.

[6] A Berenzweig, D Ellis, and S Lawrence, "Using voice segments to improve artist classification of music," *AES 22nd International Conference*, Jan 2002.

[7] B Whitman, G Flake, and S Lawrence, "Artist detection in music with minnowmatch," *Proc. of the 2001 IEEE Workshop on Neural Networks for Signal Processing*, Jan 2001.

[8] Vidya Venkatachalam, Luca Cazzanti, Navdeep Dhillon, and Maxwell Wells, "Automatic Identification of Sound Recordings," *IEEE Signal Processing Magazine*, pp. 1-8, Jul 2004.

[9] Vidya Venkatachaian, Luca Cazzanti, Kwan Fai Cheung, Navdeep Dhillon, Somsak Sukllttanon MaxWell Wells, "Automatic identification of sound recordings," 7328153, Jan 1, 2008.

[10] A Sheh and D Ellis, "Chord segmentation and recognition using EM-trained hidden Markov models," *Proc. ISMIR*, Jan 2003.

[11] K Lee and M Slaney, "Automatic chord recognition from audio using an HMM with supervised learning," *Proc. ISMIR*, Jan 2006.

[12] S Van De Par, M McKinney, and A Redert, "Musical key extraction from audio using profile training," *Proc.* of *the 7th Int. Conf. on Music* Information *Retrieval*, Jan 2006.

[13] T. Fujishima, "Realtime chord recognition of musical sound: a system using common lisp music," *ICMA, editor, International Computer Music Conference*, pp. 464–467, 1999.

[14] D Ellis and G Poliner, "Identifying 'cover songs' with beat-synchronous chroma features," *MIREX*, Jan 2006.

[15] L. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition," *Proc. IEEE* , vol. 77, no. 2, pp. 257-286, february 1989.

[16] Temple F. Smith and Michael S. and Waterman, "Identification of Common Molecular Subsequences," *Journal of Molecular Biology*, pp. 195–197, 1981.

[17] J Bello, "Audio-based cover song retrieval using approximate chord sequences: Testing shifts, gaps, swaps and …," *Proc. Int. Symp. on Music Information Retrieval*, Jan 2007.

[18] W Tsai, H Yu, and H Wang, "A query-by-example technique for retrieving cover versions of popular songs with similar melodies," *Proc. Int. Symp. on Music Information Retrieval*, Jan 2005.

[19] F Soulez, X Rodet, and D Schwarz, "Improving polyphonic and poly-instrumental music to score alignment," *Proc. ISMIR*, Jan 2003.

[20] D. P. W. Ellis. (2007) The "covers80" cover song data set.