Indexing with Musical Events and Its Application to Content-Based Music Identification

Sheng GAO*, Chin-Hui LEE†, and Qi TIAN*

*Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore, 119613
†School of Electrical & Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
{gaosheng,tian}@i2r.a-star.edu.sg  chl@ece.gatech.edu

ABSTRACT
In this paper a musical event based indexing approach is proposed and its application to content-based music identification is studied. The events, which function as term words used in text retrieval or basic speech units in speech recognition, are inferred using an unsupervised learning algorithm. Its differences with the existing methods are in that the learned low-level musicology knowledge and model selection technique are exploited to extract musical events. Our experimental analyses on a task of music identification demonstrate that the proposed indexing method is efficient, compact and robust. Using a collection of 20-second query segments on the evaluation set, the equal error rate reaches 1.57%. For applications that demand fewer false alarms, we could operate the system at a reduced false acceptance rate of 0.57% while increasing the false rejection rate to 4.58%.

1. Introduction

Content-based music identification (or audio fingerprint) is a task of identifying and verifying the identity of a given excerpt of music from a music database. It has raised much interest recently because of its potential applications in music copyright protection, automatic music library organization, broadcast monitoring, etc. [2,3,4,5,7,8]. A review on state-of-the-art audio fingerprint can be found in [8]. The concerned issues in the technology are compact and efficient representation of music content, music modeling, and music database search and organization. Here we will focus our study on the first two issues.

Signal processing techniques, such as MFCC [5], power spectrum output from filter-banks [7], MCLT [2], spectral flatness measure and spectral crest factor [4], have been applied to extract the features. In [3] beat and rhythm envelope are extracted with the beat and onset detection algorithms. Generally speaking, music content is modeled by a set of fingerprints and their relations. The fingerprints may be formed by concatenating continuous music frames [2,7], or by clusters grouped with some statistical modeling techniques, e.g. an ergodic HMM [5]. Linear discriminant analysis (LDA) can also be applied to further reducing the indexing dimension [2]. Since the fingerprints are extracted without using any musicological knowledge, there is much inconsistency and the results are usually very sensitive to time shift, which is a critical issue [8]. Moreover the fingerprints have no meaning related to common knowledge in musicology, and therefore affect their efficiency and robustness.

In this paper we apply an unsupervised musical event detection algorithm proposed in [10] to learn musical fingerprints. The musical event detector is based on combining beat and onset detection [9] and model selection [10]. The detected onset is a piece of low-level information about music structure, and can be treated as a synchronous signal for further segmentation. This makes these events more consistent and robust when compared to fixed-window or HMM based segmentation techniques [2,4,5,7]. Music passages can further be modeled with these music events and their correlations.

2. Musical event based indexing

Highly structured musical signal inspires us to investigate more efficient and compact representations by exploiting redundancy in music. However, this perceptual structure is not expressed explicitly as other signals, such as text or video. The fact that non-professional listeners can perceive and capture some structures and particular music events from signal indicates that it is possible to uncover such structures using the machine learning approaches. Here we are interested in the data-driven approaches to automatically infer some music structure and then detect coherent musical events. It is accomplished in two steps: 1) segmenting the signal using its beat onset [9] and 2) grouping these segments to find the musical events [10].

2.1 Beat onset based segmentation

The beat and onsets are the pieces of low-level information that a listener can perceive. The detected onsets can be treated as an asynchronous signal for further inferring high-level knowledge. Our experience in spectrogram reading indicates that the positions of the beat onsets for drum music show obvious differences in spectral content when comparing with their previous and following musical contexts. We expect that we can also detect these onset positions by identifying significant variations for drum-less music.
Many beat and onset detection algorithms have been proposed. Here we apply a maximum a posteriori (MAP) adaptive learning approach [9]. One good property of the algorithm is that the estimated beat and onset are more robust and consistent and less variant by integrating the learned knowledge on the beat from the previous excerpt. Given a piece of music and the frame-based feature sequence extracted from it, a temporal window is used to analyze its beat. For the sub-sequence in the window, the MAP algorithm is used to estimate the beat period based on the likelihood of the observed sub-sequence and the posterior probability of the beat period estimated from the previous excerpt. An EM algorithm is then adopted to determine the optimal beat period. The likelihood is computed based on a linear regression generative model. And the posterior probability is evaluated with a sigmoid function to smooth the likelihood of the previous excerpt. Its onsets can be further found by searching the position with the maximal energy in each beat period. These onsets are the boundaries of segmentation. This method exploits the beat information from the music signal. Its various granularity with the tempo makes it different from the fixed-size window segmentation. Moreover, these segments have somewhat perceptual meaning, especially for drum music. Figure 1 shows an example of the onset-based segmentation, where the vertical lines are the onset positions.

Figure 1 The onsets and detected musical events for "pop superstition" (X: time (s), Y: frequency, Vertical line: onset position, "A" or "B": musical event) [10]

### 2.2 Musical event detection

Many repetitive segments are often observed due to the highly structured nature of music. To improve efficiency, an unsupervised top-down clustering algorithm is used to group similar segments together and then model selection is exploited to determine the optimal cluster number. Each segment obtained above is a sub-sequence of the frame-based features. Their mean and covariance are summarized to characterize the segment. Assuming a Gaussian distribution for the features in the segment, the Kullback-Leibler (KL) distance is defined to measure the similarity between any pair of segments,

$$
    d(m_i, \Sigma_i; m_j, \Sigma_j) = \frac{1}{2} \left( \text{tr}(\Sigma_j^{-1} \Sigma_i) + \text{tr}(\Sigma_i^{-1} \Sigma_j) + (m_i - m_j)^T (\Sigma_i^{-1} + \Sigma_j^{-1}) (m_i - m_j) - D \right)
$$

where \((m_i, \Sigma_i)\) and \((m_j, \Sigma_j)\) are the means and diagonal covariances for the \(i\)-th and \(j\)-th segments, respectively, and \(D\) is the dimension of the feature vector.

With the KL distance measure, all segments are grouped into any interested number of the musical events using a top-down \(k\)-means clustering technique [10]. It is a common sense that there is the significant difference in the complexity of the music structure. Some pieces of music are simple, with only some repeats of a few events, while others are complex where diverse chords and rhythms exist. Here a Bayesian information criterion (BIC) based model selection technique is exploited to determine the optimal set of musical events.

Assume that at least \(\min C\) events are needed to sufficiently represent a piece of music and the maximal event number is set to \(\max C\). In this paper they are equal to 2 and 20 respectively. For each music, a set of models, \(\Phi = \{\phi(n)\}_{n \in [\min C, \max C]}\), is estimated, each corresponding to a possible cluster number. \(\phi(n) = [\mu(n,i), \Sigma(n,i)] \in [1,n]\) is a model with \(n\) clusters represented by \(n\) means, \(\mu(n,i)\), and covariances, \(\Sigma(n,i)\). Then the BIC is to score these models using the penalized likelihood as,

$$
    BIC(n) = -L(n) - \frac{1}{2} \kappa \cdot Q(n)
$$

where the first term in the right side, \(L(n)\), is the summary of all intra-cluster similarities, \(\kappa\) is a penalty weight (here set to 1.0), and \(Q(n)\) is a measure of the complexity of the model. Because a diagonal covariance is used in our case, \(Q(n)=2n*D*log(N)\). The optimal model, \(\phi(n^*)\), is obtained by searching the set of the models, \(\Phi\),

$$
    \phi(n^*) = \max_{\phi(n) \in \Phi} BIC(n)
$$

### 2.3 Indexing music with musical events

For any piece of music, a set of musical events will be detected using the unsupervised event detection approach described above. Each event is a partial representation of the characteristics of the music signal. And the event sequence can be used to characterize its structure (See Figure 1 for two events, “A” and “B”). Like a text document which is described by a set of term words and their correlations, a piece of music can be modeled by these detected musical events and their correlations. In this paper, we characterize each musical event with its mean and covariance and ignore their temporal information and the correlations among the events. This event-based music indexing method is attractive in three facets. First, it is compact and efficient because redundancy is greatly reduced by clustering. Second, it is robust because some low-level musicological knowledge learned from the data (i.e. beat and onsets) is exploited to synchronize the segmentation of music. In some cases it is superior to the fixed-window based segmentation. Finally the events serve as the basic units, with which more complex models can be exploited to integrate the temporal variations and correlations of music.
3. Music identification

Musical identification is to verify whether a query excerpt is in a music database and to categorize its identity once it is found in the database. Given a database with \( K \) pieces of music, each piece is modeled by a set of musical events detected with the algorithms discussed in Section 2. Here the music in the database is referred to as a target. Otherwise it is called an impostor. Denote the \( k \)-th target model by \( \Psi_k = \{ \mu_i^k, \Sigma_i^k \mid 1 \leq i \leq N_k \} \), where \( N_k \) is the event number. The query excerpt is divided into a segment sequence, \( q = (q_1, \ldots, q_m) \), using the onset-based segmentation introduced in Section 2.1. Then their means and covariances, \( \{ \mu_{q_i}, \Sigma_{q_i} \mid 1 \leq i \leq m \} \), are calculated. Its similarity, \( L_k \), with the \( k \)-th target model, can be obtained.

\[
L_k = \min_{j=1}^{N_k} d(q_j, \mu_{q_i}^k, \Sigma_{q_i}^k) \tag{4}
\]

To verify the identity, an impostor model is trained for each target music model. Here we use a simple cohort model to simulate the impostor models. First \( J \) best competitive candidates to the \( k \)-th target, \( I_k = \{ I_{k1}, \ldots, I_{kJ} \} \), is found based on the training database, where \( I_{ki} \) is the index of the target music models. The score from the \( k \)-th impostor model, \( LB_k \), can be calculated from these candidates as:

\[
LB_k = \frac{1}{J} \left( \sum_{i=1}^{J} L_{I_{ki}} \right)^{-\eta} \tag{5}
\]

where \( \eta \) is a positive constant for smoothing the scores.

Because the similarity score is non-negative, Eq. (5) is equal to the minimal score in \( J \) candidates when \( \eta \) goes to infinity. It can model the similarity distribution for the impostors. This function is proved to work well in minimum classification error based training for speech recognition [1,11]. And the verification will be done with:

\[
\begin{align*}
L_k &< LB_k \quad \text{accept} \\
L_k &\geq LB_k \quad \text{reject}
\end{align*}
\]

The evaluation method adopted here is similar to the commonly used approach in TREC speaker verification [6]. The performance is measured by the equal error rate (EER), the miss (false reject) error rate (\( E_{\text{miss}} \)), false alarm (false accept) error rate (\( E_{\text{fa}} \)), and the combined measure, detection cost function (DCF), \( C_{\text{det}} \) [6].

\[
C_{\text{det}} = c_{\text{miss}} \cdot E_{\text{miss}} \cdot P_{\text{target}} + c_{\text{fa}} \cdot E_{\text{fa}} \cdot (1 - P_{\text{target}}) \tag{7}
\]

where \( c_{\text{miss}} \), \( c_{\text{fa}} \) are the costs of a miss and a false alarm, respectively, and \( P_{\text{target}} \) is the priori probability of a target. In TREC speaker verification evaluation, they are set to 10, 1, and 0.01, respectively. In addition, the Detection Error Trade-off (DET) curve is often plotted to show and compare the performance of the identification systems.

4. Experimental setup and analysis

Now we will first describe the design of the music database for training and testing. Then the experimental results for our proposed music identification system are presented and analyzed.

4.1 Setup of training and evaluation databases

The music database consists of 807 pieces of music (i.e. 807 targets) with an average length of about 240 seconds. Diverse genres (e.g. popular, Chinese classical, songs by various singers, etc), and various formats and encoding rates (e.g. MP3, RM, etc.) are covered. All pieces of music are converted to the standard mono wave format with 16-bits resolution and 8-kHz sampling rate.

The evaluation dataset is constructed by randomly selecting some fixed-length excerpts from the music database, with a total of 3 excerpts for each piece of music. To test the effect of the granularity of music query, 3 evaluation datasets are designed for excerpts with three various durations, 5s, 10s and 20s. We call them Eval-5, Eval-10, and Eval-20, respectively. Therefore, for each evaluation dataset, there are 2,421 query excerpts. For each query, 1 target trial is tested on the music model from which the query comes from and then another 10 impostor trials are tested on randomly chosen music models that the query does not belong to. So there are totally 2,421 target queries and 24,210 impostor queries for each set. This is similar to the design rule for setting up the evaluation set in the TREC speaker verification task [6].

4.2 Performance of the event-based system

In this section we present the experimental results of the event-based verification system. First we study the effects on the performance of the candidate number, \( J \), and the smoothing coefficient, \( \eta \) in Eq.(5). Figure 2 and Figure 3 show their DET curves in the different cases, respectively. In each case, the decision points for its ERRs (labeled by a diamond) and minimal DCF (labeled by a circle) are pointed out respectively in the curves. Figure 2 clearly shows that the performance is improved while the candidate number increases from 5 to 30 ( \( \eta \) is hold constant). The same improvement is also observed from Figure 3 while \( \eta \) varies from 10 to 100. The optimal result is observed in the case of \( \eta \) equal to 100 and \( J \) equal to 30. It has a ERR 1.57% with the minimal DCF of 0.0102, in case of \( E_{\text{miss}} = 4.58\% \), and \( E_{\text{fa}} = 0.57\% \). Hereafter, all experiments use this setting for \( \eta \) and \( J \).

4.3 Effect of model selection

To show the efficiency of the model selection, we compare the performance of the event-based system with and without the model selection. For the system without
the model selection, its event number is fixed when training the music models using the clustering technique. Here it is set to 15 to make it close to the average event number (~14.1) with the model selection. Figure 4 displays their DET curves. The minimal DCF without the model selection is 0.0119, with $E_{\min} = 5.62\%$, and $E_e = 0.63\%$. And its ERR is 2.19%. So the model selection contributes to about 28.3% reduction in ERR and about 14.3% reduction for the minimal DCF.

4.4 Effects of query granularity

In this section the effects on the performance of the granularity are studied based on the sets, Eval-5, Eval-10, and Eval-20. Their comparisons are shown in Figure 5. It is clearly observed that the performance is significantly improved with the size of the granularity increasing. Their minimal DCFs are 0.0234 (Eval-5), 0.0144 (Eval-10), and 0.0102 (Eval-20), respectively.

Table 1 Effects of music editing on the accuracy

<table>
<thead>
<tr>
<th></th>
<th>Eval-20org</th>
<th>Eval-20A</th>
<th>Eval-20B</th>
<th>Eval-20C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acc</td>
<td>0.95</td>
<td>0.95</td>
<td>0.95</td>
<td>0.95</td>
</tr>
</tbody>
</table>

4.5 Robustness to music editing

Robustness to music editing means that the music should be identified even if it is modified by the various processing, e.g. equalization, filtering, stretching, etc. To this property of our identification system, a set, Eval-20org, is designed by randomly choosing 20 query excerpts from Eval-20. Then they are modified using the CoolEdit software with the following editing: 1) Eval-20A: Amplitude dynamic range processing with preset 3:1 Compressor > 30dB, 2) Eval-20B: Time stretch with slow down preset (ratio 67, overlapping 33%), and 3) Eval-20C: Time stretch with speed up preset (ratio 140, overlapping 27%). These three cases of distortion have no effects on the accuracy of identification (See Table 1). It shows the proposed method is very robust. Further more processing will be evaluated.

5. Conclusion

In this paper we propose a musical event based indexing approach and study its application in music identification. The detected events function as the term words used in text document retrieval or the basic speech units in speech recognition. Advanced statistical modeling techniques can be further exploited to model these events, although only a single Gaussian model is used in this paper. Its difference with the conventional methods is that we apply the learned low-level musicology knowledge (beat and onset) and model selection to the segmentation and the detection of the musical events. The experimental analyses on the task of music identification demonstrate that the indexing method is efficient, compact and robust. Using a collection of 20-second query segments on the evaluation set, the equal error rate reaches 1.57%. We could also operate the system at a false rejected rate of 4.58% while reducing the false accepted rate to 0.57%.

REFERENCES