MUSIC SUMMARIZATION USING KEY PHRASES

*Beth Logan and Stephen Chu**

Compaq Computer Corporation
Cambridge Research Laboratories
One Kendall Square, Building 700, 2nd Floor
Cambridge, Massachusetts 02139
United States

ABSTRACT

Systems to automatically provide a representative summary or 'Key Phrase' of a piece of music are described. For a 'rock' song with 'verse' and 'chorus' sections, we aim to return the chorus or in any case the most repeated and hence most memorable section. The techniques are less applicable to music with more complicated structure although possibly our general framework could still be used with different heuristics.

Our process consists of three steps. First we parameterize the song into features. Next we use these features to discover the song structure, either by clustering fixed-length segments or by training a hidden Markov model (HMM) for the song. Finally, given this structure, we use heuristics to choose the Key Phrase. Results for summaries of 18 Beatles songs evaluated by ten users show that the technique based on clustering is superior to the HMM approach and to choosing the Key Phrase at random.

1. INTRODUCTION

As the magnitude and use of multimedia databases grows, efficient ways to automatically find the 'gist' of the contents become necessary. We investigate the problem of automatically summarizing music, specifically songs of 'rock' or 'pop' genre. Potential applications of automatic music summarization include multimedia indexing, multimedia data searching, content-based music retrieval, and online music distribution.

An intuitive approach to the problem of music summarization is to first automatically generate the score (musical transcription) for the song and then look for repetitive patterns or motifs in the melody. Indeed much previous work has investigated automatic music transcription. However, although there are a number of well-understood techniques for monophonic transcription (pitch tracking), only limited success has been achieved for polyphonic music [1], [2]. Thus reliably finding the melody in a complex arrangement is difficult or impossible using present technologies. Moreover, pattern discovery in temporal sequences is also a difficult problem, let alone the fact that the sequence may well be noisy due to variations in the melody.

* Stephen Chu is a PhD student at the University of Illinois, IL, USA. This work was performed during a summer internship at Cambridge Research Laboratories.

Recent work suggests that the acoustic structure of music is more important than its written form [3]. Human listeners hear groups of notes or chords as single objects in many circumstances. Our music summarization techniques are therefore based on detecting broad spectral changes rather than attempting to track the melody. The assumption is that 'interesting' parts of the song will reoccur and be spectrally similar. It should be noted that although this assumption is reasonable for rock or folk music, it may be less applicable to classical music.

We parameterize each song using 'Mel-cepstral' features that have found great success in speech processing applications [4]. They have also been used with success for music retrieval [5].These give a smoothed version of the magnitude spectrum of short ($<$ 1sec) sections of the audio signal. Thus even if the melody changes slightly or new instruments are introduced, the features will be somewhat robust to these changes.

Given these features for a song, we use various clustering techniques to discover the song structure. We then use heuristics to extract the Key Phrase given this structure.

This paper is organized as follows. In Section 2, we describe the summarization procedure. Experiments and results are given in Section 3 and conclusions in Section 4.

2. SUMMARIZATION PROCEDURE

Our summarization process consists of three main steps. First, as described in the previous section, we characterize each song by a sequence of features. Second, we label subsequences of the song in order to discover interesting structure. We use two techniques to determine the labels: top-down clustering and unsupervised learning of hidden Markov models (HMMs). Finally, given the song structure, the Key Phrase is chosen. This process is shown in Figure 1. We describe each of these steps in detail in the following sections.

2.1. Calculation of Mel-Cepstral Features

For each song to be summarized, we calculate a sequence of Mel-cepstral features. Such a parameterization has proven highly successful in in speech recognition applications [4]. The cepstral features are calculated as follows.

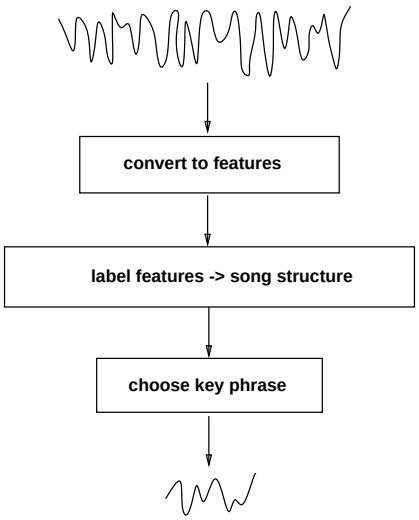

Figure 1: Top level diagram of the process of creating a Key Phrase from a song

First the audio signal is divided into fixed length and possibly overlapping segments called 'frames'. For each frame, the log of the power spectrum is calculated. Next, the spectrum is warped according to the so-called Mel scale which emphasizes perceptually important lower frequencies. Finally, the discrete cosine transform is taken of the resulting vector. This approximates the Karhunen-Loeve transform for the Mel-spectral features, resulting in a vector of effectively decorrelated cepstral coefficients. Typically a subset of these coefficients (often 13) are used as features.

2.2. Discovery of Song Structure

The second step of the summarization process is to label each frame of the song such that frames which are similar have the same label. For example, the introduction, verse and chorus of a song would ideally be assigned different labels. Songs with more complicated structures would have more labels as required. We describe here two techniques that differ in the degree to which the song is modeled, and the distance measure used to compare the features.

2.2.1. Clustering

The clustering technique uses bottom-up clustering to group similar cepstral features. It operates as follows.

1. Divide the sequence of features for the song into fixed length contiguous segments. These segments are the initial pool of clusters.
2. Assuming the features in each cluster have a Gaussian distribution, calculate the mean and covariance for each cluster.
3. Compute and store the distortion between each pair of clusters. We use a modified cross-entropy or Kullback Leibler (KL) distance described below.

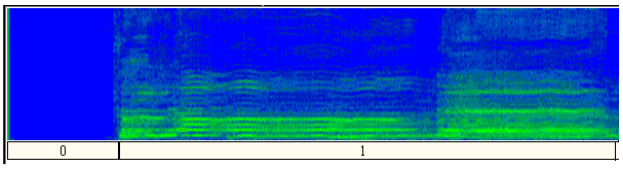

Figure 2: Examples of labels '0' and '1' assigned to a song spectrum

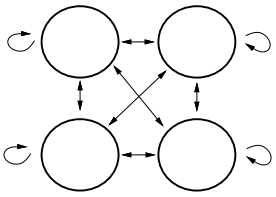

Figure 3: Structure of a 4 state ergodic HMM. The figure shows the possible transitions between states

4. Pick the pair of clusters with the lowest distortion between them.
5. If this is less than a predefined threshold, combine these two clusters and go to step 2.
6. Each distinct cluster is now assigned a label. All the frames in this cluster are given this label. A typical assignment of labels might be as shown in Figure 2.

We use a modified KL distance, KL2, to compare clusters as described in [6]. We use

$$KL2(A;B) = KL(A;B) + KL(B;A) \quad (1)$$

where A and B are two distributions and

$$KL(A;B) = E_A\{\log(pdf(A)) - \log(pdf(B))\}. \quad (2)$$

Assuming A and B are Gaussian distributions

$$KL2(A;B) = \frac{\Sigma_A}{\Sigma_B} + \frac{\Sigma_B}{\Sigma_A} + (\mu_A - \mu_B)^2 \cdot \left(\frac{1}{\Sigma_A} + \frac{1}{\Sigma_B}\right). \quad (3)$$

2.2.2. HMM Approach

The HMM approach seeks a statistical model that reflects the structure of the data. Instead of arbitrarily dividing the data, we attempt to learn the segmentation from the data itself.

We model each song by an ergodic HMM. This is a fully connected finite-state machine. The structure of a 4-state model is shown in Figure 3. Each state is modeled by a Gaussian distribution. The parameters of the model are transition probabilities between states and the means and variances for each state.

We use unsupervised Baum-Welsh training to train the HMM given the sequence of cepstral features for the song. After training, we use Viterbi decoding to determine the most likely state for each frame. This state gives the label

for each frame. An excellent description of HMMs including how to train their parameters and how to use Viterbi decoding to determine the most likely state for each frame is given in [4].

2.3. Choosing the Key Phrase

The result of the preceding steps is a labeled version of the song to be summarized as shown in Figure 2. We now use this implied song structure to choose the Key Phrase.

Various heuristics can be applied at this stage. We assert that the most interesting or memorable part of the song will be that which occurs most frequently. We thus first determine the most frequently occurring label. For example, for the song shown in Figure 2 this would be label '1'.

We then choose the Key Phrase as a fixed length segment, say 10 seconds, from among the frames with the most frequent label. Again various heuristics can be applied. In this work, we take the longest section containing the most frequent label which occurs in the first half of the song. We restrict our search space to the first half of the song to avoid choosing long instrumental sections which generally do not provide good summaries.

3. EXPERIMENTS

We conduct user tests to evaluate our music summaries since there is no ground truth. We first conduct tests with one user to determine various parameter settings. We then conduct a final evaluation with a set of different users.

We confine the scope of our experiments to songs of rock or pop genre. In preliminary informal experiments, we noted that users found it difficult to assess the quality of a song summary unless they had some knowledge of the song. Therefore, for the general user tests we used a pool of 18 Beatles songs which had all been Number 1 hits in the US. We chose these because the Beatles are one of the most successful rock groups of all time and many people have been exposed to their music. Our aim is to to introduce some objectivity into our choice of test set rather than selecting songs arbitrarily.

3.1. Parameter Selection

Before conducting the final user test, we informally investigate a number of parameterization variations. Our conclusions are based on the summaries of 50 Beatles songs evaluated by one of the authors. These songs are well known to the author and do not include the 18 songs used in the final evaluation. For each song a 10 second summary was generated and given a rating of 'good', 'average' or 'poor'. For comparison purposes, these are enumerated '3', '2' and '1' respectively.

The parameter space to be investigated is reasonably large. For the clustering technique, we can vary the size of the initial segments and the threshold of the stopping criteria. For the HMM approach, the number of states is the main parameter. Additionally, there are many variations in the way the cepstral features used by both approaches can be calculated. We can vary the cepstral analysis order, the range of the spectrum over which the cepstrum is calculated, the spectrum window size and the frame rate.

Threshold	Average Score
0.2	2.3
0.4	2.4
0.6	2.2
1.0	1.8

Table 1: Average scores for 50 Beatles song summaries determined using the clustering technique and evaluated by one user as a function of stopping criteria

Number of States	Average Score
5	2.0
7	2.0
10	2.1
12	2.1

Table 2: Average scores for 50 Beatles song summaries determined using the HMM technique and evaluated by one user as a function of number of states

Given then the size of the parameter space and the fact that even informal tests are time consuming, we did not experiment a great deal with the cepstral feature parameterization. We instead use 'standard' speech recognition cepstral features. These are somewhat optimized for human listening and very optimized for speech.

Specifically, we use audio sampled at 16kHz divided into 25.6ms windows at a 10ms frame rate. The spectrum of each frame is band-limited to 133Hz-6855Hz. A Mel-scaled filter bank consisting of 40 bandpass filters is then used to calculate 13 cepstral coefficients. Although it is possible that a variation on these settings would be more suitable for music summarization, such experimentation is beyond the scope of this paper. We did not however notice any spectacular improvement in performance for very limited experiments in which these parameters were varied.

For the clustering technique, the size of the initial segments controls the resolution of the result since all final segments will be at least this long. It also controls the computational complexity since the distortion must be calculated between each pair of segments for clustering. We set this size to one second which gives a good trade-off between these factors.

We then investigate the choice of stopping threshold. We experimented with a threshold which was the ratio of maximum to minimum distortion for all current clusters. Table 1 shows the average score obtained for the summaries of the 50 Beatles songs for various thresholds using the clustering technique. Given these results, we use a stopping threshold of 0.4 for the general user tests.

For the HMM approach we use a fully connected ergodic HMM. We experiment with varying numbers of states. Table 2 shows the average scores obtained for these tests. We see that 10 or 12 states would be an appropriate choice for this data.

However, the table does not show the fact that as the number of states increases, less data is available to train the parameters of each state. We alleviate this problem some-

Summarization Technique	Average Score	Variance
Random	1.9	0.7
Clustering	2.4	0.6
HMM	2.1	0.5

Table 3: Average scores with their variances for various summaries of 18 Number 1 Beatles songs evaluated by ten users

what by forcing all states to share the same covariance matrix. However for some songs there is still insufficient data which leads to numerical problems. This problem arises for 1 song in the 10 state system and 4 songs in the 12 state system. Ideally, we would dynamically choose the number of HMM states for each song using a HMM structure learning technique such as [7]. However, for the work in this paper, we compromise and use a 7 state HMM to model each song.

3.2. General User Tests

Using these parameter settings, we now conduct general user tests on the summaries of 18 Number 1 Beatles songs. For each song, we generate three 10-second summaries: one using the clustering method; another using the HMM approach and a third taken randomly from the song. For each song, the user is told the song title and is then presented with the three summaries. These are presented in a random order and the user is not told which technique is used to generate each.

Ten users participated in the evaluation. Again, songs are ranked as 'good', 'average' or 'poor' which we again ennumerate at 3, 2 and 1 respectively. Additionally, a song can be skipped if the user does not feel familiar enough with it to make a judgment. In practice, there were 29 instances of users not knowing songs out of a possible 180 tests.

The average scores with variances for each summarization technique are listed in Table 3. From these results, we see that the clustering method achieved better than random performance for this task. The performance is significantly better than random at 95% confidence. However, the performance of the HMM method was not significatly better than random.

Note that the random technique did not fare poorly. This is because the Beatles songs are fairly repetitive so the chance of hitting a 'good' excerpt in a random segment is reasonably high. However, although the HMM method and the random method have similar performance, the former consistently starts a summary at more natural places such as the beginning of a phrase.

We did not discuss with the users how they should evaluate the summaries before they performed the tests. Afterward however we asked them what criteria they used. The following points were highlighted:

- it is desirable to have the song title sung in the summary;
- a vocal portion is better than an instrumental one;
- it is preferable to start at the beginning of a phrase rather than in the middle.

The first two points imply that the results could be improved by incoporating singing detection into the system. This could be used as an additional criteria when choosing the Key Phrase given the song structure.

4. CONCLUSIONS

This work addresses the problem of automatic music summarization. We investigate two approaches to finding keyphrases based on clustering segments and learning HMMs. Subjective experiments on Beatles songs show that the clustering approach is significantly better than using random summaries. The HMM method is capable of uncovering structures in music but was not better than random.

The structure discovery potential of the HMM approach is limited by the inflexibility of a fixed model topology. Future work should therefore investigate using HMM structure learning techniques such as [7]. Additionally, since users preferred Key Phrases which contained ideally the sung song title or at least a vocal portion, both the clustering and HMM approach could benefit by the incorporation of singing detection.

Finally, we would like to stress that further work is required to identify how generic our summarization techniques are. We tested with Beatles songs because these are well known. Future work would include investigation of other song genres.

5. ACKNOWLEDGMENTS

We would like to thank members of the Speech Group at CRL for useful discussions and also the users who participated in the evaluations.

6. REFERENCES

[1] K. D. Martin, "Automatic transcription of simple polyphonic music: robust front end processing," tech. rep., M. I. T., 1996.

[2] K. Kashino and H. Murase, "Music recognition using note transition context," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1998.

[3] K. D. Martin, E. D. Scheirer, and B. L. Vercoe, "Music content analysis through models of audition," in *Proc. ACM Multimedia Workshop on Content Processing of Music for Multimedia Applications*, 1998.

[4] L. R. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*. Prentice–Hall, 1993.

[5] J. T. Foote, "Content-based retrieval of music and audio," in *SPIE*, pp. 138–147, 1997.

[6] M. A. Siegler, U. Jain, B. Raj, and R. M. Stern, "Automatic segmentation, classification and clustering of broadcast news data," in *DARPA Speech Recognition Workshop*, pp. 97–99, 1997.

[7] M. Brand, "Pattern discovery via entropy minimization," in *Proceedings of Uncertainty '99*, 1999.