Audio Retrieval by Rhythmic Similarity

Jonathan Foote
FX Palo Alto Laboratory, Inc.
3400 Hillview Ave.
Building 4
Palo Alto, CA 94304 USA
foote@fxpal.com

Matthew Cooper
FX Palo Alto Laboratory, Inc.
3400 Hillview Ave.
Building 4
Palo Alto, CA 94304 USA
cooper@fxpal.com

Unjung Nam
CCRMA
Department of Music
Stanford University
Stanford, CA 94305 USA
unjung@stanford.edu

ABSTRACT

We present a method for characterizing both the rhythm and tempo of music. We also present ways to quantitatively measure the rhythmic similarity between two or more works of music. This allows rhythmically similar works to be retrieved from a large collection. A related application is to sequence music by rhythmic similarity, thus providing an automatic "disc jockey" function for musical libraries. Besides specific analysis and retrieval methods, we present small-scale experiments that demonstrate ranking and retrieving musical audio by rhythmic similarity.

1. INTRODUCTION

We present audio analysis algorithms that can automatically rank music by rhythmic and tempo similarity. Music in a user's collection is analyzed using the "beat spectrum," a novel method of automatically characterizing the rhythm and tempo of musical recordings [1]. The beat spectrum is a measure of acoustic self-similarity as a function of time lag. Highly repetitive music will have strong beat spectrum peaks at the repetition times. This reveals both tempo and the relative strength of particular beats, and therefore can distinguish between different kinds of rhythms at the same tempo. Unlike previous approaches to tempo analysis, the beat spectrum does not depend on particular attributes such as energy, pitch, or spectral features, and thus will work for any music or audio in any genre. The beat spectrum is calculated for every music file in the user's collection. The result is a collection of "rhythmic signatures" for each file. We present methods of measuring the similarity between beat spectra, and thus between the original audio. Given a similarity measure, files can be ranked by similarity to one or more selected query files, or by similarity with any other musical source from which a beat spectrum can be measured. This allows users to search their music collections by rhythmic similarity, as well an enable novel applications. For example, given a collection of files, an application could sequence them by rhythmic similarity, thus functioning as an "automatic DJ."

2. BEAT SPECTRAL SIMILARITY

Details of the beat-spectral analysis are presented in [1]; we include a short version here for completeness. The beat spectrum is calculated from the audio using three principal steps. First, the audio is parametrized into a spectral or other representation. This results in a sequence of feature vectors. Second, a distance measure is used to find the similarity between all pairwise combinations of feature vectors, hence times in the audio. This is embedded into a two-dimensional representation called a similarity matrix. The beat spectrum results from finding periodicities in the similarity matrix, using diagonal sums or autocorrelation.Given two works, we can compute two beat spectra $B_1(l)$ and $B_2(l)$; both are 1-dimensional functions of lag time *l*. In practice, *l* is truncated to some number L of discrete values. This yields L-dimensional vectors, from which the Euclidean or other distance functions can be computed.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page.
© 2002 IRCAM – Centre Pompidou

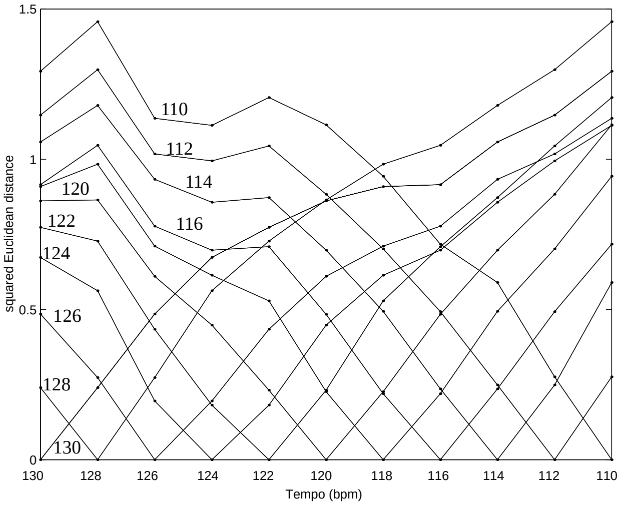

Figure 1. Euclidean Distance vs. Tempo

2.1 Experiment 1

In this experiment, we determine how well Euclidean distance between beat spectra measures tempo difference. We generated different-tempo versions of the identical musical excerpt using commercially available music editing software, which can change the duration of a musical waveform without altering the pitch. This musical excerpt consists of 16 4/4 bars of live vocals and instrumentals over a rhythmic beat. The original excerpt was played at 120 beats per minute (bpm; also denoted MM). Ten tempo variations were generated at 2 bpm intervals from 110 to 130 bpm. [1]

A first test of measuring beat spectral difference is a simple Euclidean distance between beat spectra. To this end, beat spectra were computed for each excerpt, and the squared Euclidean distance computed for all pairwise combinations. Figure 1 shows the result. Each line shows the Euclidean distance between one source excerpt and all other files. This graphically demonstrates that the Euclidean distance increases relatively monotonically for increasing tempo differences. Of course, this is a highly artificial case in that examples of the same music at different tempos are relatively rare. Still, it serves as a "sanity check" that the beat spectrum does in fact capture useful rhythmic information.

2.2 Experiment 2

The corpus for this experiment are 10-second excerpts taken from the audio track of "Musica Si," available as item V22 of the MPEG-7 Content Set [2]. There were four songs that were long enough to extract multiple ten-second samples. Each song is repre-

[1] This audio data can be found at http://www.fxpal.com/people/foote/musicr/tempor.html

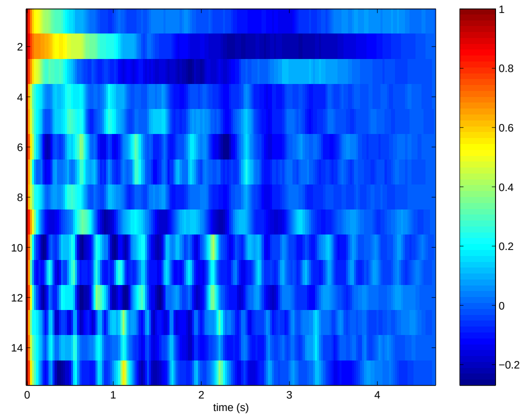

Figure 2. Beat spectra of retrieval data set.

sented by three ten-second excerpts, save for a pop/rock song whose chorus and verse are each represented by three excerpts respectively. Each excerpt was assumed to be relevant to other excerpts from the same tune, and not relevant to all other excerpts. The one exception is that the verse and chorus of the pop/rock song were markedly different in rhythm and so are assumed to not be relevant to each other. Thus we have three ten-second excerpts from each of five relevance classes (three songs plus two song sections), for a total of 15 excerpts as shown in Table 1.

Each beat spectrum was normalized by scaling so the peak magnitude (at zero lag) was unity. Next, the mean was subtracted from each vector. Finally the beat spectra were truncated in time. Because the short-lag spectra is similar across all files and thus not informative, the first 116 ms was truncated, as well as lags longer than 4.75 s.t The result was a zero-mean vector having a length of 200 values, representing lags from 116 ms to 4.75 s for each musical excerpt. (The effect of varying the truncation regions was not examined, and it is unlikely that the ones chosen are optimal.)

Several distance measures were investigated. First, each of the 15 corpus documents was then ranked by similarity to each query using Euclidean distance. Each query had 2 relevant documents in the corpus, so this was the cutoff point used to measure retrieval precision. After ranking by increasing Euclidean distance, 24 of the 30 possible documents were relevant, giving a retrieval precision of 80%. A second measure used is a cosine metric, or the inner product of the vectors normalized by the product of their magnitudes. This measure performed significantly better than the Euclidean distance: 29 of the 30 documents retrieved were relevant, giving a retrieval precision of 96.7% at this cutoff.

The final distance measure is based on the Fourier coefficients of the beat spectrum. The fast Fourier transform was computed for each beat spectral vector. The log of the magnitude was then determined, and the mean subtracted from each coefficient. Because high “frequencies” in the beat spectra are not rhythmically significant, the transform results were truncated to the 25 lowest coefficients. The cosine distance metric was computed for the 24 zero-mean Fourier coefficients, which served as the final distance metric. Experimentally, this measure performed identically to the cosine metric using an order of magnitude fewer parameters.

3. CONCLUSION

Table 1. Retrieval set: 10-second excerpts from “Musica Si” [2]

Time (mm:ss)	Song Title (approximate)	Description	Set
09:12	“Toto Para Me”	acoustic guitar + vocals	A
09:02	“Toto Para Me”	acoustic guitar + vocals	A
08:52	“Toto Para Me”	acoustic guitar + vocals	A
07:26	“Never Loved You Anyway”	pop/rock chorus	B
06:33	“Never Loved You Anyway”	pop/rock chorus	B
06:02	“Never Loved You Anyway”	pop/rock verse	C
05:52	“Never Loved You Anyway”	pop/rock verse	C
05:30	“Never Loved You Anyway”	pop/rock chorus	B
04:53	“Never Loved You Anyway”	pop/rock verse	C
01:39	“Everybody Dance Now”	dance + rap vocals	D
01:29	“Everybody Dance Now”	dance + rap vocals	D
01:19	“Everybody Dance Now”	dance + vocals	D
00:25	“Musica Si Theme”	theme + vocals	E
00:15	“Musica Si Theme”	theme + vocals	E
00:05	“Musica Si Theme”	theme intro	E

We have presented an approach to finding rhythmically similar music and audio. Though the experiments here are on an admittedly very small corpus, there is no reason that these methods could not be scaled to thousands or even millions of works.

In contrast to other approaches, the beat spectrum does not depend on assumptions such as silence, periodic peaks, or particular time signatures in the source audio. Practical applications include an “automatic DJ” for personal music collections, and we are currently prototyping such a system. We hypothesize that a satisfying sequence of arbitrary music can be achieved by minimizing the beat-spectral difference between successive songs. This ensures that song transitions are not jarring, for example following a particularly slow or melancholic song with a rapid or energetic one. Additionally, rhythmic retrieval could usefully be with other systems that retrieve music by pitch or timbral similarity [7]. Such a hybrid retrieval engine might allow users to trade off spectral and rhythmic similarity to suit their particular information needs.

4. REFERENCES

[1] Foote, J. and Uchihashi, S. “The Beat Spectrum: A New Approach to Rhythm Analysis,” in *Proc. International Conference on Multimedia and Expo* 2001. http://www.fxpal.com/people/foote/papers/ICME2001.htm

[2] MPEG Requirements Group. “Description of MPEG-7 Content Set,” http://www.darmstadt.gmd.de/mobile/MPEG7/Documents/N2467.html

[3] Scheirer, E., “Tempo and Beat Analysis of Acoustic Musical Signals,” in *J. Acoust. Soc. Am.* **103**(1), Jan. 1998, pp 588-601.

[4] G. Tzanetakis, P. Cook, “Automatic Musical Genre Classification of Audio Signals,” in *Proc. ISMIR 2001*

[5] Wold, E., Blum, T., Keislar, D., and Wheaton, J., “Classification, Search and Retrieval of Audio,” in *Handbook of Multimedia Computing*, ed. B. Furht, pp. 207-225, CRC Press, 1999.

[6] Cliff, David, “Hang the DJ: Automatic Sequencing and Seamless Mixing of Dance Music Tracks,” *HP Technical Report HPL-2000-104*, Hewlett-Packard Labs, Bristol UK

[7] Pye, D., “Content-based Methods for the Management of Digital Music,” in *Proc. ICASSP 2000*, vol. IV pp 2437.