A Multifaceted Approach to Music Similarity

Kurt Jacobson
University of Miami
Coral Gables, FL USA
kurtj@miami.edu

Abstract

Previous work has explored the concept of music similarity measures and a variety of methods have been proposed for calculating such measures. This paper describes a system for music similarity which attempts to model and compare some of the more musically salient features of a set of audio signals. A model for timbre and a model for rhythm are implemented directly from previous work, and a model for song structure is developed. The different models are weighted and combined to provide an overall music similarity measure. The system is tested on a small set of popular music files spanning eleven different genres. The system is tuned to estimate genre boundaries using multidimensional scaling – a technique that allows for quick visualization of similarity data. An "automatic DJ" application, that generates playlists based on the music similarity models, serves as a subjective evaluation for the system.

Keywords: music similarity, automatic DJ, playlist generation, multidimensional scaling, song structure.

1. Introduction

From purchase to playback, digital audio files are becoming a ubiquitous part of the music consumption process. At all levels, improved methods for navigating collections of digital music files are desired. Audio-based music similarity measures could be applied to this retrieval problem in a number of ways including playlist generation, recommendation of unknown pieces or artists, organization and visualization of music collections, and retrieval by example.

Recent work suggests that the limits of what can be achieved with audio-based music similarity measures are close at hand. Some of the most advanced work in music similarity suggests there exists a "glass ceiling" limiting the effectiveness of this approach [1]. Combining additional dimensions of similarity, such as the temporal loudness descriptor used in [2], provides only a small benefit. Music similarity is undeniably complex, and modeling it is likely to require a high-dimensional space. Perhaps using half-a-dozen descriptors as opposed to two or three descriptors will provide some improvement. Perhaps the number is much higher or even non existent.

The system described in this work uses prior art to model the rhythm [5] and timbre of music signals [3]. A song structure model is also developed, although its effectiveness is questionable.

The system is tested on a set of popular songs from the iTunes music store. Multidimensional scaling (MDS) is used to visualize the songs in the test set and to estimate how well the music similarity measures identify genre boundaries.

2. Related Work

There has been a significant amount of research on music similarity and even more research on audio-based genre classification [1-3, 9, 11]. Both areas of research use some type of content-based descriptors extracted from audio signals.

Mel frequency cepstral coefficients (MFCC) have been used in previous work to determine spectral similarity [1-4]. The MFCC frames for a given music signal are grouped into clusters and then some statistical technique is used to compare cluster models between songs. This technique is most closely associated with the timbral attributes of a music signal and is used here as the timbre model.

A variety of methods have also been developed for describing and comparing the rhythmic similarity for a set of music [5, 9]. The approach developed in [5] is used here. It provides information about both rhythm and tempo, making it more appropriate for an automatic DJ application.

MDS is applied to the similarity results as a means to quickly estimate system performance. This approach has been applied to musical timbre perception research in [8], but it is usually not applied to music similarity.

3. System Design

A timbre, rhythm, and song structure model are extracted for a given music file and stored in XML format. The XML files associated with a given set of music files are then used to calculate inter-song similarities. MDS is then applied to visualize a "music space," providing estimation of how well the similarity data follows genre boundaries. The similarity data is also applied to playlist generation.

3.1 Timbre Model

The timbre model is based on [1-4] and uses the open source MA Matlab toolbox. The timbre model is based on the k-means clustering of MFCC, following the approach outlined in [3]. Although [2] suggests Gaussian mixture models provide improved results, the Earth Mover's Distance is used to compare timbre models.

3.2 Rhythm Model

The rhythm model uses the approach described in [5]. A self-similarity matrix is calculated for a portion of the audio signal. The same MFCC frames are used as in the timbre model. Summing across the super-diagonals of the self-similarity matrix generates a "beat spectrum" vector, which is stored in XML. The vectors are compared using a cosine distance.

3.3 Song Structure Model

To extract the song structure model a low-resolution version of the self-similarity matrix is calculated. By correlating a Gaussian-tapered checkerboard kernel with the main diagonal of the self-similarity matrix, a "novelty index" is calculated. This process has been applied to automatic audio segmentation and is described in detail in [6].

A threshold is set for the novelty index. Every positive cross of the threshold is considered to indicate a significant change in the music. For every positive cross, a counter increments and the position of the change relative to the song length is recorded. This thought process is described in pseudo code in equation (1).

$$\begin{array}{l} if\,(Nv_p(i) > Nv_{threshold}\ \&\ Nv_p(i-1) \le Nv_{threshold}) \\ \{C_p = C_p + 1 \\ rl_p(j) = i / length(Nv_p) \\ j = j + 1\} \end{array} \qquad (1)$$

Where $Nv_{threshold}$ is some constant and $Nv_p(i-1)$ refers to the previous value of the novelty index for song p. This way, only the positive crossings of the $Nv_{threshold}$ increment C_p. This is a fairly accurate method for finding changes in an audio signal. Note that rl_p records the normalized locations of the changes. Should $rl_p = 0.5$, this would indicate a change in the middle of the song. The mean of rl_p is taken to get μl_p.

To compare the structure model of song p with that of song q, C_p and C_q are considered magnitudes, while μl_p and μl_q are taken to be the corresponding angles. The song structure similarity between songs p and q is calculated as a Euclidean distance between the resulting vectors:

$$SM_{structure}(p,q) = \sqrt{\begin{array}{l} \left(C_p \cos(\sigma \cdot \mu l_p) - C_q \cos(\sigma \cdot \mu l_q)\right)^2 \\ + \left(C_p \sin(\sigma \cdot \mu l_p) - C_q \sin(\sigma \cdot \mu l_q)\right)^2 \end{array}} \qquad (2)$$

Here, σ represents some maximum angle. The lower σ is set, the less impact the relative location of changes has on structure model similarity. Preliminary tests indicate that $\sigma = \pi/4$ is an appropriate value. This allows songs with different change distributions, but the identical numbers of changes to still be similar. $SM_{structure}$ is normalized to one by dividing by the maximum value of $SM_{structure}$. This normalization conforms to the other similarity measures.

3.4 Combined Similarity

To derive one matrix of similarity distances SM_{total} that combines all the similarity models the following equation is used:

$$SM_{total} = w_{timbre}SM_{timbre} + w_{rhythm}SM_{rhythm} + w_{structure}SM_{structure} \qquad (3)$$

The weights reflect the relative importance of each model to overall music similarity. The weight values should sum to one. This is a very simple method for combining these measures, and a more advanced method maybe called for in future testing.

4. Testing the System

4.1 iTunes top tens

To create a test pool of digital music files, the top ten rated songs in eleven different genres were purchased from the iTunes music store.

The genres were selected arbitrarily and included Hip-hop/Rap, Classical, Pop, World, Jazz, Dance, Electronic, Country, Blues, Alternative, and R&B/Soul.

The iTunes top ten ratings are based on route sales data. To date, over 980 million songs have been purchased since the service first launched on April 28, 2003. There are currently iTunes stores available in 21 countries. Given the popularity and broad scope of the iTunes music store, the genre distinctions applied to the test songs can hardly be ignored, even if they are questionable. A useful system should at least loosely identify genre boundaries defined by the iTunes music store.

This sampling of the iTunes content creates a test pool with distinct files that represent the same song title. This type of duplication occurs when a song title is in the top ten lists for two genres, when a song title has both radio edit and explicit versions in a top ten list, or when a song title has both album and single versions in a top ten list.

4.2 Specific genre sets

Selections from the digital music collections of three professional DJs in the Miami area were also tested. Each collection represents a specific genre. These genres include drum and bass, hip hop, and reggae-dancehall.

4.3 Multidimensional Scaling

As a means to quickly calibrate the weights applied to each similarity model, MDS solutions were calculated and visualized. MDS is a set of statistical techniques that allow for the visualization of data based on distances or dissimilarity data. The details of MDS are given in [7]. A two-dimensional metric MDS is applied to the music similarity data. In the visualization, songs are color-coded by genre to estimate genre classification. The visualization of the MDS analysis seemed to indicate the best genre separation in the iTunes test set with weight values of $w_{timbre} = 0.6$, $w_{rhythm} = 0.3$, and $w_{structure} = 0.1$

Using MDS to estimate the performance of a music similarity system is not a common practice. However, the technique has been applied to timbre perception research [8]. It also provides a quick estimate of system performance without formally applying the genre classification problem.

4.4 The Auto DJ

As a subjective evaluation of the system, an automatic DJ application generates playlists based on the music similarity data generated by the system. Because the system is sensitive to tempo, no time-stretching or pitch shifting is applied to the music signals. Instead, quick fades are applied at musical changes indicated by the stored novelty indexes associated with the music signals. The user can alternatively prompt the auto DJ to mix to the next song immediately. More sophisticated mixing approaches are also implemented.

5. Results

While a rigorous quantitative evaluation of this system's performance is not provided, there are some encouraging results. As described in section 4.1, in the iTunes test set, there exist several song titles that appear twice. The system identifies all eight pairs of doubles as being most similar to the alternate version. The auto DJ application also consistently plays these double songs back-to-back.

A similar result was found when applying the system to the reggae-dancehall set. In this genre, it is common for several different vocalists to use identical backing tracks to create distinct songs. Songs with the same backing track are said to be on the same "riddim" (from rhythm). The auto DJ consistently plays songs on the same riddim back-to-back.

6. Future work

A more rigorous testing of this system is required. Using the ISMIR contest music collections as a test set and applying the system to automatic genre classification would allow for a quantitative comparison to other music similarity systems.

Additional models should be added to the system as well. Some model for melodic or harmonic similarity could improve system performance.

Also, the models currently in the system could be improved. The current rhythm model has no significant psychoacoustic basis. Different methods for the characterization of music based on rhythmic patterns have been developed such as [9] and should also be explored in the context of this system.

Perhaps most importantly, the relevance of the song structure model should be rigorously evaluated. Currently, the model is only justified by a few test cases and the MDS estimations. Closely examining the structure models of several test songs seems to indicate that the current method is good at identifying break-down sections in dance music or hip hop, but less effective at identifying more subtle changes, like those found in jazz.

7. Acknowledgments

This work makes use of the open source MA Toolbox by E. Pampalk [13].

References

[1] E. Pampalk, A. Flexer, and G. Widmer. "Improvements of Audio-Based Music Similarity and Genre Classification," *Proc ISMIR*, 2005.

[2] J.-J. Aucouturier and F. Pachet. "Improving Timbre Similarity: How high is the sky?" *JNRSAS*, 2004.

[3] B. Logan, "A Content-Based Music Similarity Function," *Cambridge Research Laboratory Technical Report Series*, 2001.

[4] A. Berenzweig, B. Logan, D. Ellis, and B. Whitman. "A Large-Scale Evaluation of Acoustic and Subjective Music Similarity Measures." *Johns Hopkins University*, 2003.

[5] J. Foote, M. Cooper, and U. Nam. "Audio Retrieval by Rhythmic Similarity." *Proc ISMIR*, 2002.

[6] J. Foote and M. Cooper. "Media Segmentation using Self-Similarity Decomposition." *FX Palo Alto Laboratory*, 2002.

[7] J. Kruskal and M. Wish. "Multidimensional Scaling." *Sage University, Quantitative Applications in the Social Sciences*, 1976.

[8] J. Grey. "Multidimensional Perceptual Scaling of Musical Timbres." *Journal of the Acoustical Society Of America*, 1977.

[9] S. Dixon, F. Gouyon, and G. Widmer. "Towards Characterization of Music Via Rhythmic Patterns." *Proc ISMIR*, 2004.

[10] G. Tzanetakis, G. Essl, and P. Cook. "Automatic Musical Genre Classification of Audio Signals." *Proc ISMIR*, 2001.

[11] T. Lidy and A. Rauber. "Evaluation of Feature Extractors and Psychoacoustic Transformations for Music Genre Classification." *Proc ISMIR*, 2005.

[12] D. Ellis, B. Whitman, A. Berenzweig, and S. Lawrence. "The Quest for Ground Truth in Musical Artisit Similarity." *Proc ISMIR*, 2002.

[13] E. Pampalk. "A Matlab Toolbox to Compute Music Similarity from Audio." *Proc ISMIR*, 2004.