A COMPARISON OF SIGNAL-BASED MUSIC RECOMMENDATION TO GENRE LABELS, COLLABORATIVE FILTERING, MUSICOLOGICAL ANALYSIS, HUMAN RECOMMENDATION, AND RANDOM BASELINE

Terence Magno
Cooper Union
magno.nyc@gmail.com

Carl Sable
Cooper Union
CarlSable.Cooper@gmail.com

ABSTRACT

The emergence of the Internet as today's primary medium of music distribution has brought about demands for fast and reliable ways to organize, access, and discover music online. To date, many applications designed to perform such tasks have risen to popularity; each relies on a specific form of music metadata to help consumers discover songs and artists that appeal to their tastes. Very few of these applications, however, analyze the signal waveforms of songs directly. This low-level representation can provide dimensions of information that are inaccessible by metadata alone. To address this issue, we have implemented signal-based measures of musical similarity that have been optimized based on their correlations with human judgments. Furthermore, multiple recommendation engines relying on these measures have been implemented. These systems recommend songs to volunteers based on other songs they find appealing. Blind experiments have been conducted in which volunteers rate the systems' recommendations along with recommendations of leading online music discovery tools (Allmusic which uses genre labels, Pandora which uses musicological analysis, and Last.fm which uses collaborative filtering), random baseline recommendations, and personal recommendations by the first author. This paper shows that the signal-based engines perform about as well as popular, commercial, state-of-the-art systems.

1 INTRODUCTION

The nature of online music distribution today is characterized by massive catalogs of music unbounded by physical constraints. As pointed out in [1], current technology has offered music listeners "massive, unprecedented choice in terms of what they could hear". The number of songs available on-line is in the billions, and many millions of users are continuing to flock from traditional means of obtaining music (e.g., CD stores) to online alternatives [11].

With such a vast amount of music available on the Internet, end users need tools for conveniently discovering music previously unknown to them (whether recently released or decades old). In the context of electronic music distribution, it is the goal of today's online discovery tools to automatically recommend music to human listeners. This is no simple task; a program must have an automated way of computing whether or not one song is, in some sense, similar to some other set of songs (i.e., to songs that are already liked by the user to whom the program is recommending new music). In accordance with this goal, we have designed and implemented three systems that use signal-based music similarity measures to recommend songs to users.

In this paper, we first discuss existing methods of automatic music recommendation, including a discussion of commercial, state-of-the-art systems that use them, in Section 2. Next, we discuss techniques for automatically computing signal-based music similarity, including a description of our own similarity measures, in Section 3; the optimization of the measures is discussed in Section 4. In Section 5, we discuss how these similarity measures have been used to design and implement three automatic, signal based music recommendation engines. Section 6 describes experiments in which volunteers have rated the recommendations of these systems, along with those of the popular systems described in Section 2, a baseline system, and human recommendations. We evaluate the results of these experiments in Section 7. We then state some general conclusions in Section 8.

2 STRATEGIES FOR AUTOMATIC MUSIC RECOMMENDATION

Three possible strategies of automatic music recommendation involve expert opinions, collaborative filtering, and musicological analysis. Recommendation by expert opinion often relies on the application of genre labels to songs and artists. The wide variety of music genre labels has arisen through a multifaceted interplay of cultures, artists, music journalists, and market forces to make up the complex hierarchies that are in use today [16]. Currently, the largest database of music that is organized by genre is Allmusic [1],

[1] http://www.allmusic.com/

where professional editors compose brief descriptions of popular musical artists, often including a list of similar artists [6]. In the context of automatic music recommendation, recent research has effectively pointed out significant deficiencies of the traditional genre labeling methodology. For one, as discussed in [16], there is no general agreement in the music community as to what kind of music item genre classification should be consistently applied: a single song, an album, or an artist. Second, as discussed in [14], there is no a general agreement on a single taxonomy between the most widely-used music databases on the Internet. Lastly, it is noted in [16] that the criteria for defining music genres have, for countless years, been inconsistent; some labels are geographically defined, some are defined by a precise set of musical techniques, while others arise from the lexical whims of influential music journalists.

Given these inconsistencies, musicological analysis aims to determine music similarity in a way that transcends conventional genre labels, focusing primarily on music theoretic description of the vocal and instrumental qualities of songs. This technique was spearheaded by the Music Genome Project (MGP) in 2000, whose research culminated in the music discovery website/tool Pandora [2] [10]. The automatic recommendation algorithm behind Pandora involves comparisons of very particular descriptions of songs. The description process involves analysis of songs by a team of professional music analysts, each song being represented by about 150 "genes," where each gene describes a musicological quality of the song. Perhaps the most apparent drawback of musicological analysis — especially in the context of Pandora — is that while the recommendation process is automated, the description aspect is not. It is this aspect that contributes to the relatively slow rate at which new content is added to the Pandora database.

Also designed within the context of online music discovery, collaborative filtering works according to the principle that if songs or artists you like occur commonly in other users' playlists, then you will probably also like the other songs or artists that occur in those playlists. According to [8], "if your collection and somebody else's are 80% alike, it's a safe bet you would like the other 20%". One of the most popular on-line recommendation engine to use collaborative filtering is Last.fm [3], which boasts 15 million active users and 350 million songs played every month [12]. One problem with collaborative filtering systems is that they tend to highlight popular, mainstream artists. As noted in [8], Last.fm "rarely surprises you: It delivers conventional wisdom on hyperdrive, and it always seems to go for the most obvious, common-sense picks." In other words, collaborative filtering is not helpful for discovering lesser known music which a user might highly appreciate.

The past several years have seen considerable progress in the development of mathematical methods to quantify musical characteristics of song waveforms based on the content of their frequency spectra. In particular, these methods have enabled the extraction of features of a song's waveform that are correlated with the song's pitch, rhythmic, and timbral content. Timbre can be said to be the most important of these three elements when subjectively assessing musical similarity between a pair of songs; indeed, it may even be said that the global timbral similarity between two pieces of music is a reasonable — and often sufficient — estimate of their overall musical similarity [4].

These research efforts have also gone on to evaluate and test several different timbre-based music similarity measures applied to a number of signal-based music information retrieval tasks, including supervised and unsupervised classification of entire music databases and the segmentation and summarization of individual songs [16]. Following the lead of these efforts, we have applied signal-based measures of music similarity to the task of automatic recommendation of music. An automatic recommendation engine built on a signal-based music similarity measure would possess the advantages that current online music discovery tools merely trade off. It would boast the ability to describe and compare pieces of music based purely on their musical qualities, and would also facilitate the rapid addition of new content to a music database that does not require human intervention.

3 COMPUTING MUSIC SIMILARITY

Our signal based recommendation engines rely on the ability to automatically compute the similarity of two songs. First, the relevant information about each song — features — is computationally derived from its waveform data. Second, a compact representation of the song is obtained by modeling the distribution of its feature data using mixture and clustering algorithms. Third, a metric for comparing mixture models of songs is used to estimate the similarity between the feature distributions of two different songs. In effect, the timbral similarity between the two songs is mathematically computed.

As a whole, this music similarity measure framework allows a user to present a song query to the signal-based recommendation engine and receive a set of song recommendations (i.e., similar songs) drawn from a target music database. The similarities of the recommended songs to the query song are determined via signal processing alone, without human intervention. In this section, a general overview is given of the three similarity measures examined in this paper. Our implementations of these measures are based partly on those proposed in [9, 15, 17].

The music feature dataset extracted by the measures' analysis front-ends are the Mel-frequency cepstral coefficients (MFCC's). These perceptually-motivated features capture the "spectral shape" — and effectively, the timbral quality

[2] http://www.pandora.com/
[3] http://www.last.fm/

— of a music signal within a small frame of the waveform [18, 6]. In the literature, the MFCC feature set has already shown effective performance for various audio classification experiments [6, 18, 13, 3].

3.1 K-Means Clustering with Earth Mover's Distance

The first similarity measure used in our work was originally proposed by Logan and Salomon in their work in [13]. In this architecture, K-means clustering of a target song's feature vectors is performed during the statistical modeling stage, with each data cluster being subsequently fit with a Gaussian component to form a Gaussian Mixture Model (GMM). Also in line with what was proposed in [13], the distance metric stage of the first similarity measure incorporates the Earth Mover's Distance (EMD). The EMD expands the Kullback-Leibler divergence — a distance metric for comparing individual probability distributions — to the comparison of mixtures of distributions (in this case, GMM). For the remainder of the paper, this similarity measure combining K-means training of GMM's with the Earth Mover's Distance for GMM comparison is referred to by the shorthand term *KM+EMD*.

3.2 Expectation-Maximization with Monte Carlo Sampling

The second similarity measure that we have relied on uses the Expectation-Maximization (EM) algorithm to train the parameters of each GMM component. Aucouturier and Pachet introduced and refined the use of EM to model music feature distributions in [2, 3, 4]. This method makes use of vectors sampled directly from the GMM's of the two songs to be compared; the sampling is performed computationally via random number generation. This sampling process corresponds roughly to recreating a song from its timbre model [4], and is known as Monte Carlo Sampling (MCS). Using MCS in conjunction with GMM training via Expectation-Maximization is in line with what was originally proposed by Aucouturier and Pachet in [2, 3, 4]. For the remainder of the paper, the similarity measure based on this approach is referred to as *EM+MCS*.

3.3 Average Feature Vector with Euclidean Distance

In the early work of Tzanetakis and Cook [18], a simple way is presented to construct an averaged vector representation of a song's MFCC's. They propose that low-order statistics such as mean and variance should be calculated over segments called texture windows that are more meaningful perceptually. With respect to human auditory perception, the length of a so-called texture window roughly corresponds to the minimum duration of time required to identify a particular sound or music "texture" that corresponds to its overall timbral character. This has led us to test a simpler similarity measure which does not involve the training of a GMM. For each song, a single "average feature vector" is constructed from means and variances taken across the texture windows of the song's waveform. The song's representative vector may then be compared to that of another song by taking the Euclidean distance between them. The similarity measure based on this approach is referred to as *AV+EUC*.

4 PARAMETER OPTIMIZATION

In order to use any of the similarity measures discussed in Section 3, the values of several parameters must be selected. Perhaps most importantly are the dimensionality of the MFCC vectors (N) and the number of Gaussian components in a GMM (M). The parameter M is not applicable when using AV+EUC as a similarity measure. Other parameters include the sampling frequency of the song waveforms (f_s), the frame length (N_f), and for the case of EM+MCS, the distance sample rate (N_{DSR}). It has been hypothesized in [4] that these later three parameters are independent of N and M, and we have decided to use the values that were obtained in [4] and [5]; namely, f_s = 44,100 Hz (44.1 kHz), N_f = 1,102 samples (corresponding to a frame duration of 25 ms), and N_{DSR} = 2,000.

In order to optimize the first two parameters, two authors of this paper have subjectively evaluated the similarity of 200 song pairs that were randomly selected from a corpus containing approximately 10,000 songs spanning 40 different genres. Each author has rated each pair of songs using a one to four scale explained in Table 1; half ratings (e.g., 2.5) were also allowed. For the similarity measures KM+EMD and EM+MCS, N was varied from 5 to 25 in steps of 5, and M was varied from 5 to 30 in steps of 5. For the similarity measure AV+EUC, N was taken from the set {3, 4, 5, 8, 10, 15, 20, 23, 30, 40, 50}.

Two-fold cross validation has been used to evaluate each parameter configuration. The 200 song pairs are randomly divided into two disjoint subsets with 100 song pairs each. Similarity measures are computed for the each of the first 100 song pairs; these pairs are then sorted according to their similarities and grouped into ten bins with ten song pairs each. Each bin is then labeled with an average rating, according to the authors, of the ten songs in the bin, rounded to the nearest 0.5. Next, similarity measures are computed for the other 100 song pairs, and each is assigned a rating according to the bin from the first 100 song pairs into which the current song pair would fall. These automatically assigned ratings for the second subset of 100 songs are used to compute the average computer-to-human correlation for the current parameter configuration. The entire process is then repeated, swapping the two subsets of 100 songs, and the two correlations computed for each parameter configuration are averaged together. Correlation has been used as defined in [7].

Rating	Meaning	Description
4	"Extremely Similar"	If a person likes one of the songs, it would be rare they wouldn't like the other.
3	"Similar"	If a person likes one of the songs, it's fairly likely they would like the other.
2	"Not Similar"	Liking the first song does not increase or decrease the chances of liking the other.
1	"Totally Different"	It is highly unlikely that a person would like both songs at the same time.

Table 1: Subjective scale for rating music similarity.

The optimal values of N and M for KM+EMD were N=20 and M=15 leading to a computer-to-human correlation of 0.496. The optimal values of N and M for EM+MCS were N=5 and M=25 leading to a computer-to-human correlation of 0.547. The optimal value of N for AV+EUC was 4 leading to a computer-to-human correlation of 0.484. According to [7], these numbers represent medium to large correlations. Note that the correlation between the two authors was 0.613, a large correlation, and since it is unlikely that an automatic similarity measure would outperform humans, this could be considered a reasonable upper bound on the achievable correlations. For the remainder of the paper, the optimized configurations of the three approaches are referred to as *KM+EMD(20,15)*, *EM+MCS(5,25)* and *AV+EUC(4)*.

5 SIGNAL-BASED MUSIC RECOMMENDATION

Three self-sufficient music recommendation engines have been implemented, each incorporating one of the three optimized music similarity measures. The same corpus of 10,000 songs mentioned in Section 4 serves as the source from which each engine draws its recommendations. Each engine accepts a single music query from a user in the form of a digital file. The song's MFCC features are then extracted, and a representative mixture model or vector is computed from the feature distribution.

To begin the recommendation process, the distance between the query song model and the model of each song in the target music corpus is computed, resulting in a total of approximately 10,000 distances generated for the query. Since the corpus is organized by album, the songs in each album are then arranged in order of least to greatest distance from the query song (i.e., "most timbrally similar" to "least timbrally similar"). The most similar song is then chosen from each album; in this way, we are not allowing two recommendations from the same album. The representative songs from each album are sorted in order of least to greatest distance from the query, and three songs are selected at random from the top 2% of songs in the sorted list. During this process, any song selection bearing the same artist as one of the previous selections is discarded, and random selection is repeated as necessary. The final three song selections are considered to be the recommendations based on the query song.

The authors found it justified to present the user with only one song from each artist according to the reasonable assumption that most artists' songs are, generally speaking, timbrally and stylistically consistent. In the case that many of an artist's songs are computed to be extremely close to a query song, the respective artist would be overrepresented in the resulting recommendations. It suffices to assign the role of "gateway song" to the closest song from such an artist to introduce a user to the artist and their discography, and it gives other possibly relevant songs the chance to find a place in the recommendation set.

6 EXPERIMENTAL DESIGN

We have conducted experiments to compare our recommendation engines to today's leading online music discovery tools (i.e., Pandora, Last.fm, and Allmusic). 15 volunteers not involved with the research were recruited, and each volunteer was requested to submit one song based on which three recommendations would be generated by every system. Each volunteer was instructed to verify that Pandora.com, Last.fm, and Allmusic.com all recognize the title and artist of their chosen song prior to submission. The 15 submitted songs were well distributed in terms of their Allmusic genre classification — one belonged to the *Proto-Punk* genre, one to *Hip-Hop*, one to *Hard Rock*, one to *MPB* (Música Popular Brasileira), one to *Jazz*, one to *Alternative Rock*, one to *Indie Pop*, two to *Punk Rock*, two to *Classical*, and three to *Rock-n-Roll*.

To generate recommendations from Pandora, the title of the volunteer's song was submitted to the Pandora website, and the first three songs returned were used (unless any single artist happened to be repeated, in which case the latter song by the artist would be skipped and the next song by a new artist was used in its place). To generate recommendations from Last.fm, which uses artists as opposed to songs to generate suggestions, the artist of the volunteer's song was submitted and the first three recommended songs (also excluding songs from identical artists) were used. To generate recommendations from Allmusic, three songs were randomly chosen from the same narrow genre as the volunteer's submission (not allowing duplicate artists). As a baseline, we also created a simple engine that randomly chooses three songs from the entire corpus (not allowing duplicate artists). As an upper bound, the first author of this paper suggested three personal recommendations.

The three systems described in Section 5, the three online discover tools, the baseline system, and the first author each generated three recommendations based on every submitted song, so 24 total recommendations were generated for each volunteer. These recommendations were returned to the volunteer in a randomized order, without indicating which recommendation was produced by which method; in

the rare instance that multiple engines would choose the same song, that song would only be included in the list once. Each volunteer was then asked to rate each recommendation on a one to five scale explained in Table 2; half ratings were also allowed.

Rating	Description
5	"A top-notch recommendation"
4	"A good recommendation"
3	"An OK recommendation"
2	"Not a good recommendation"
1	"A very bad recommendation"

Table 2: Subjective scale for rating recommendations.

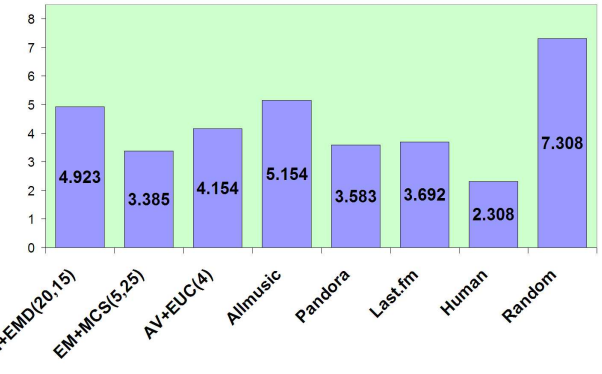

Figure 2: Average rankings of all music recommendation engines, computed across the entire set of volunteers.

7 RESULTS AND EVALUATION

Ultimately, 13 of the 15 volunteers submitted their subjective ratings of the recommendations for their query song. For each volunteer, the performance of each of the eight recommendation engines have been assessed by computing the average of the ratings given to the three songs recommended by that particular engine. These averages have also been used to determine the rank (from first to eighth place) of each engine; engines which tied were assigned equal ranks. To evaluate the performance of all eight recommendation methods across the entire set of volunteers, the ratings and rankings assigned by all volunteers for each method have been averaged; the results are shown in Figure 1 and Figure 2.

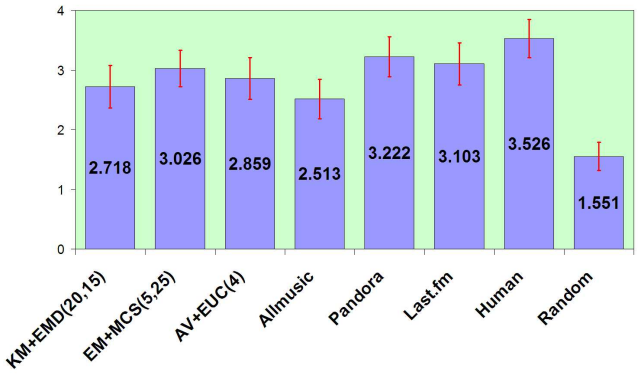

Figure 1: Average ratings for all music recommendation engines, computed across the entire set of volunteers.

It can be seen from Figures 1 and 2 that all of the software-based recommendation engines significantly outperform the baseline random recommender (which received the lowest average rating and worst average rank value), but none perform quite as well as the human recommender (who received the highest average rating and best average rank). According to average ratings, the order of automatic recommendation engines, from best to worst, is Pandora, Last.fm, EM+MCS(5, 25), AV+EUC(4), KM+EMD(20,15), and Allmusic. According to average ranks, the order, from best to worst, is EM+MCS(5, 25), Pandora, Last.fm, AV+EUC(4), KM+EMD(20,15), and Allmusic.

The red bars in Figure 1 represent 95% confidence intervals, assuming normal distrubutions for ratings. Note that the confidence interval for random recommendations has no overlap with that of any other approach; the closest gap is of size approximately 0.4. With the exception of Pandora compared to Allmusic, the confidence intervals of the automatic recommendation engines all overlap, and even for the one exception, the confidence intervals miss each other by only a few hundredths of a point. The top three automated systems - Pandora, Last.fm, and EM+MCS(5, 25) - have confidence intervals that partially overlap with that of the human recommender.

It is not surprising that the two professional online music tools - Pandora and Last.fm - are rated the highest by the 13 volunteers. Note, however, that our signal based recommendation engines trail closely behind, and in particular, EM+MCS(5,25) achieves an average rating only 6% lower than that of Pandora and 2.5% lower than that of Last.fm. In fact, based on average rank, EM+MCS(5,25) performs the best of all the automated systems, indicating that although its average rating is not as high, it beats the professional systems more often than it loses to them. Among the three signal-based recommendation engines, it is not surprising that EM+MCS(5,25) performs the best. The merits of a music similarity measure that utilizes expectation-maximization and Monte Carlo sampling have already been established in the literature [2, 3, 4].

8 CONCLUSIONS

This paper has shown that a signal-based recommendation engine can perform comparably to popular, state-of-the-art commercial music discovery applications when subjected to human evaluation. This fact further highlights the important role that timbre plays in subjective judgment of music similarity; a timbre similarity measure that relies on signal analysis alone appears to be approximately as robust a musical descriptor as musicological analysis or collaborative filtering, and moreso than conventional genre taxonomies.

The results also show that music recommendations given by a fellow human do not satisfy the sensibilities of a music consumer all of the time. Accurately predicting a person's musical tastes is highly dependent on several cultural, sociological, and psychoacoustic factors. Nevertheless, it may be seen that, acting independently, each recommendation engine — whether signal-based or not — produces significantly more accurate recommendations than a baseline random recommender. We can thus say that the particular aspects of music highlighted by each recommendation method are all integral parts of whatever holistic sense of music similarity a person may be said to possess.

Sun Microsystems' Paul Lamere, who is one of the leading researchers in music information retrieval, has dubbed the ideal music discovery engine the "celestial jukebox" [8]. It may be posited this ideal hypothetical engine would be one that somehow combines all the similarity measurement techniques evaluated in this paper, and others as well. Given the positive results discussed in this paper, there is little doubt in the minds of the authors that signal-based music similarity measures will be a *sine qua non* feature of the celestial jukebox of the future.

9 REFERENCES

[1] Anderson C. "The Rise and Fall of the Hit", *Wired Magazine*, vol. 14, no. 7, July, 2006.

[2] Aucouturier, J.-J. and Pachet, F. "Finding songs that sound the same" *Proceedings of the IEEE Benelux Workshop on Model-Based Processing and Coding of Audio*, 2002.

[3] Aucouturier, J.-J. and Pachet, F. "Music similarity measures: What's the use?". *Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR)*, Paris, France, October 2003.

[4] Aucouturier, J.-J. and Pachet, F. "Improving Timbre Similarity: How high is the sky?", *Journal of Negative Results in Speech and Audio Sciences*, vol. 1, no. 1, 2004.

[5] Aucouturier, J.-J. and Pachet, F. and Sandler, M. "The Way It Sounds: Timbre Models for Analysis and Retrieval of Music Signals", *IEEE Transactions on Multimedia*, vol. 7, no. 6, December 2005.

[6] Berenzweig, A., Logan, B., Ellis, D.P.W. and Whitman, B. "A large-scale evaluation of acoustic and subjective music similarity measures.", *Proceedings of the AES 22nd International Conference on Virtual, Synthetic, and Entertainment Audio*, Espoo, Finland, June 2002.

[7] Cohen J. "Statistical Power Analysis for the Behavioral Sciences", Lawrence Erlbaum Associates, 1988.

[8] Dahlen, C. "Better Than We Know Ourselves", http://www.pitchforkmedia.com/article/feature/36524-better-than-we-know-ourselves, May, 2006. [Online; last accessed March 2008].

[9] Ellis, D.P.W. "PLP and RASTA (and MFCC, and inversion) in MATLAB". http://www.ee.columbia.edu/dpwe/resources/matlab/rastamat/, 2005. [Online; last accessed March 2008]

[10] Folini, F. "An Interview with Tim Westergren, Pandora Founder and Chief Strategy Officer", http://blog.novedge.com/2007/10/an-interview-wi.html, October, 2007. [Online; last accessed March 2008].

[11] Gupta, R. "The New, New Music Industry", http://gigaom.com/2007/02/15/the-new-new-music-industry/, February, 2007. [Online; last accessed March 2008].

[12] Lake, C. "Interview with Martin Stiksel of Last.fm", http://www.e-consultancy.com/news-blog/362081/interview-with-martin-stiksel-of-last-fm.html, November, 2006. [Online, last accessed March 2008].

[13] B. Logan and A. Salomon. "A music similarity function based on signal analysis", *Proceedings of the 2001 International Conference on Multimedia and Expo (ICME '01)*, 2001.

[14] Pachet, F. and Cazaly, D. "A taxonomy of musical genres". *Proceedings of the Content-Based Multimedia Access Conference (RIAO)*, Paris, France, 2000.

[15] Pampalk, E. "A MATLAB Toolbox to Compute Music Similarity From Audio". Technical Report, Austrian Research Institute for Artificial Intelligence, 2004.

[16] Scaringella, N., Zoia, G. and Mlynek, D. "Automatic Genre Classification of Music Content: A survey". *IEEE Signal Processing Magazine*, pp. 133-141, March 2006.

[17] Slaney, M. "Auditory Toolbox, Version 2". Technical Report, Interval Research Corporation, 1998.

[18] Tzanetakis, G. and Cook, P. "Musical Genre Classification of Audio Signals". *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, July 2002.