Automatic Musical Genre Classification Of Audio Signals

George Tzanetakis
Computer Science Department
35 Olden Street
Princeton NJ 08544
+1 609 258 5030
gtzan@cs.princeton.edu

Georg Essl
Computer Science Dep.
35 Olden Street
Princeton NJ 08544
+1 609 258 5030
gessl@cs.princeton.edu

Perry Cook
Computer Science and Music Dep.
35 Olden Street
Princeton NJ 08544
+1 609 258 5030
prc@cs.princeton.edu

ABSTRACT
Musical genres are categorical descriptions that are used to describe music. They are commonly used to structure the increasing amounts of music available in digital form on the Web and are important for music information retrieval. Genre categorization for audio has traditionally been performed manually. A particular musical genre is characterized by statistical properties related to the instrumentation, rhythmic structure and form of its members. In this work, algorithms for the automatic genre categorization of audio signals are described. More specifically, we propose a set of features for representing texture and instrumentation. In addition a novel set of features for representing rhythmic structure and strength is proposed. The performance of those feature sets has been evaluated by training statistical pattern recognition classifiers using real world audio collections. Based on the automatic hierarchical genre classification two graphical user interfaces for browsing and interacting with large audio collections have been developed.

1. INTRODUCTION
Musical genres are categorical descriptions that are used to characterize music in music stores, radio stations and now on the Internet. Although the division of music into genres is somewhat subjective and arbitrary there are perceptual criteria related to the texture, instrumentation and rhythmic structure of music that can be used to characterize a particular genre. Humans are remarkably good at genre classification as investigated in [1] where it is shown that humans can accurately predict a musical genre based on 250 milliseconds of audio. This finding suggests that humans can judge genre using only the musical surface without constructing any higher level theoretic descriptions as has been argued in [2]. Up to now genre classification for digitally available music has been performed manually. Therefore techniques for automatic genre classification would be a valuable addition to the development of audio information retrieval systems for music.

In this work, algorithms for automatic genre classification are explored. A set of features for representing the music surface and rhythmic structure of audio signals is proposed. The performance of this feature set is evaluated by training statistical pattern recognition classifiers using audio collections collected from compact disks, radio and the web. Audio signals can be automatically classified using a hierarchy of genres that can be represented as a tree with 15 nodes. Based on this automatic genre classification and the extracted features two graphical user interfaces for browsing and interacting with large digital music collections have been developed. The feature extraction and graphical update of the user interfaces is performed in real time and has been used to classify live radio signals.

2. RELATED WORK
An early overview of audio information retrieval (AIR) (including speech and symbolic music information retrieval) is given in [3]. Statistical pattern recognition based on the extraction of spectral features has been used to classify Music vs Speech [4], Isolated sounds [5, 6] and Instruments [7]. Features related to timbre recognition have been explored in [8,9]. Extraction of psychoacoustic features related to music surface and their use for similarity judgements and high level semantic descriptions (like slow or loud) is explored in [10]. Content-based similarity retrieval from large collections of music is described in [11]. Automatic beat tracking systems have been proposed in [12, 13] and [14] describes a method for the automatic extraction of time indexes of occurrence of different percussive timbres from an audio signal. Musical genres can be quite subjective making automatic classification difficult. The creation of a more objective genre hierarchy for music information retrieval is discussed in [15]. Although the use of such a designed hierarchy would improve classification results it is our belief that there is enough statistical information to adequately characterize musical genre. Although manually annotated genre information has been used to evaluate content-based similarity retrieval algorithms to the best of our knowledge, there is no prior published work in automatic genre classification.

3. FEATURE EXTRACTION

3.1 Musical Surface Features

In this work the term "musical surface" is used to denote the characteristics of music related to texture, timbre and instrumentation. The statistics of the spectral distribution over time can be used in order to represent the "musical surface" for pattern recognition purposes. The following 9-dimensional feature vector is used in our system for this purpose: (**mean-Centroid, mean-Rolloff, mean-Flux, mean-ZeroCrossings, std-Centroid, std-Rolloff, std-Flux, std-ZeroCrossings, LowEnegry**).

The means and standard deviations of these features are calculated over a "texture" window of 1 second consisting of 40 "analysis" windows of 20 milliseconds (512 samples at 22050 sampling rate). The feature calculation is based on the Short Time Fourier Transform (STFT). that can be efficiently calculated using the Fast Fourier Transform (FFT) algorithm [16].

The following features are calculated for each "analysis" window: (M[f] is the magnitude of the FFT at frequency bin f and N the number of frequency bins):

- **Centroid :** $$C = \frac{\sum_1^N fM[f]}{\sum_1^N M[f]} \quad \textbf{(1)}$$

  The Centroid is a measure of spectral brightness.

- **Rolloff :** is the value R such that :

  $$\sum_1^R M[f] = 0.85 \sum_1^N M[f] \quad \textbf{(2)}$$

  The rolloff is a measure of spectral shape.

- **Flux:** $$F = \left\| M[f] - M_p[f] \right\| \quad \textbf{(3)}$$

  where M_p denotes the FFT magnitude of the previous frame in time. Both magnitude vectors are normalized in energy. Flux is a measure of spectral change.

- **ZeroCrossings:** the number of time domain zerocrossings of the signal. ZeroCrossings are useful to detect the amount of noise in a signal.

- **LowEnergy:** The percentage of "analysis" windows that have energy less than the average energy of the "analysis" windows over the "texture" window.

3.2 Rhythm features

The calculation of features for representing the rhythmic structure of music is based on the Wavelet Transform (WT) which is a technique for analyzing signals that was developed as an alternative to the STFT. More specifically, unlike the STFT that provides uniform time resolution for all frequencies the DWT provides high time resolution for all frequencies, the DWT provides high time resolution and low frequency resolution for high frequencies and high time and low frequency resolution for low frequencies.

The Discrete Wavelet Transform (DWT) is a special case of the WT that provides a compact representation of the signal in time and frequency that can be computed efficiently. The DWT analysis can be performed using a fast, pyramidal algorithm related to multirate filterbanks [17]. An introduction to wavelets can be found in [18].

For the purposes of this work, the DWT can be viewed as a computationally efficient way to calculate an octave decomposition of the signal in frequency. More specifically the DWT can be viewed as a constant Q (bandwidth / center frequency) with octave spacing between the centers of the filters.

In the pyramidal algorithm the signal is analyzed at different frequency bands with different resolutions by decomposing the signal into a coarse approximation and detail information. The coarse approximation is then further decomposed using the same wavelet step. The decomposition is achieved by successive highpass and lowpass filtering of the time domain signal and is defined by the following equations:

$$y_{high}[k] = \sum_n x[n] g[2k-n] \quad \textbf{(4)}$$

$$y_{low}[k] = \sum_n x[n] h[2k-n] \quad \textbf{(5)}$$

where $y_{high}[k], y_{low}[k]$ are the output of the highpass (g) and lowpass (h) filters, respectively after subsampling by two. The DAUB4 filters proposed by Daubechies [19] are used.

The rhythm feature set is based on detecting the most salient periodicities of the signal. Figure I shows the flow diagam of the beat analysis. The signal is first decomposed into a number of octave frequency bands using the DWT. Following this decomposition the time domain amplitude envelope of each band is extracted separately. This is achieved by applying full wave rectification, low pass filtering and downsampling to each band. The envelopes of each band are then summed together and an autocorrelation function is computed. The peaks of the autocorrelation function correspond to the various periodicities of the signal's envelope. These stages are given by the equations:

1. **Full Wave Rectification (FWR):**

   $$y[n] = abs(x[n]) \quad \textbf{(6)}$$

2. **Low Pass Filtering (LPF):** (One Pole filter with an alpha value of 0.99) i.e:

   $$y[n] = (1-\alpha)x[n] - \alpha y[n] \quad \textbf{(7)}$$

3. **Downsampling** (↓) by k (k=16 in our implementation)**:**

   $$y[n] = x[kn] \quad \textbf{(8)}$$

4. **Normalization (NR)** (mean removal)**:**

   $$y[n] = x[n] - E[x[n]] \quad \textbf{(9)}$$

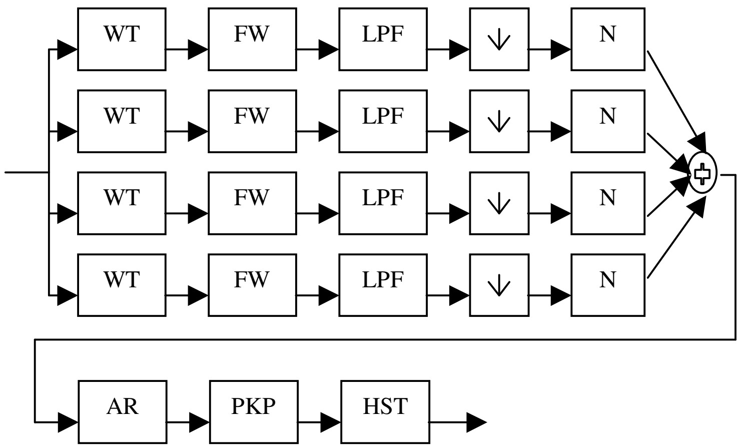

Fig. I Beat analysis flow diagram

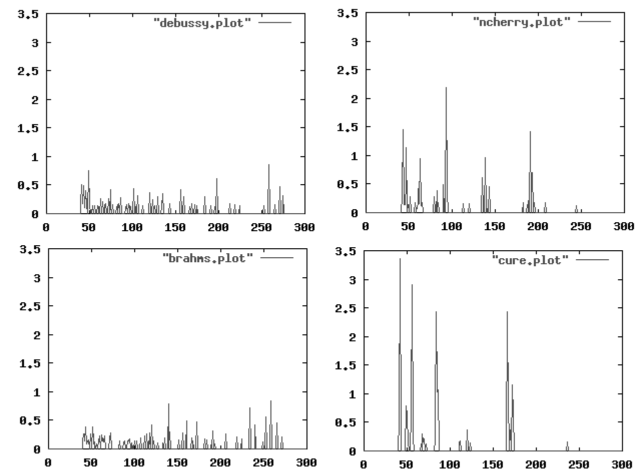

Fig. II Beat Histograms for Classical (left) and Pop (right)

5. **Autocorrelation (AR)** (computed using the FFT for efficiency) **:**

$$y[n] = \frac{1}{N}\sum_{n} x[n]x[n+k] \qquad \textbf{(10)}$$

The first five peaks of the autocorrelation function are detected and their corresponding periodicities in beats per minute (bpm) are calculated and added in a "beat" histogram. This process is repeating by iterating over the signal and accumulating the periodicities in the histogram. A window size of 65536 samples at 22050 Hz sampling rate with a hop size of 4096 samples is used. The prominent peaks of the final histogram correspond to the various periodicities of the audio signal and are used as the basis for the rhythm feature calculation.

The following features based on the "beat" histogram are used:

1. **Period0**: Periodicity in bpm of the first peak Period0
2. **Amplitude0**: Relative amplitude (divided by sum of amplitudes) of the first peak.
3. **RatioPeriod1**: Ratio of periodicity of second peak to the periodicity of the first peak
4. **Amplitude1**: Relative amplitude of second peak.
5. **RatioPeriod2, Amplitude2**, **RatioPeriod3, Amplitude3**

These features represent the strength of beat ("beatedness") of the signal and the relations between the prominent periodicities of the signal. This feature vector carries more information than traditional beat tracking systems [11, 12] where a single measure of the beat corresponding to the tempo and its strength are used.

Figure II shows the "beat" histograms of two classical music pieces and two modern pop music pieces. The fewer and stronger peaks of the two pop music histograms indicate the strong presence of a regular beat unlike the distributed weaker peaks of classical music.

The 8-dimensional feature vector used to represent rhythmic structure and strength is combined with the 9-dimensional musical surface feature vector to form a 17-dimensional feature vector that is used for automatic genre classification.

4. CLASSIFICATION

To evaluate the performance of the proposed feature set, statistical pattern recognition classifiers were trained and evaluated using data sets collected from radio, compact disks and the Web. Figure III shows the classification hierarchy used for the experiments. For each node in the tree of Figure III, a Gaussian classifier was trained using a dataset of 50 samples (each 30 seconds long). Using the Gaussian classifier each class is represented as a single multidimensional Gaussian distribution with parameters estimated from the training dataset [20]. The full digital audio data collection consists of 15 genres * 50 files * 30 seconds = 22500 seconds (i.e 6.25 hours of audio).

For the Musical Genres (Classical, Country…..) the combined feature set described in this paper was used. For the Classical Genres (Orchestra, Piano…) and for the Speech Genres (MaleVoice, FemaleVoice…) mel-frequency cepstral coefficients [21] (MFCC) were used. MFCC are perceptually motivated features commonly used in speech recognition research. In a similar fashion to the Music Surface features, the means and standard deviations of the first five MFCC over a larger texture window (1 second long) were calculated. MFCCs can also be used in place of the STFT-based music surface features with similar classification results. The use of MFCC as features for classifying music vs speech has been explored in [22].

The speech genres were added to the genre classification hierarchy so that the system could be used to classify live radio signals in real time. "Sports announcing" refers to any type of speech over noisy background.

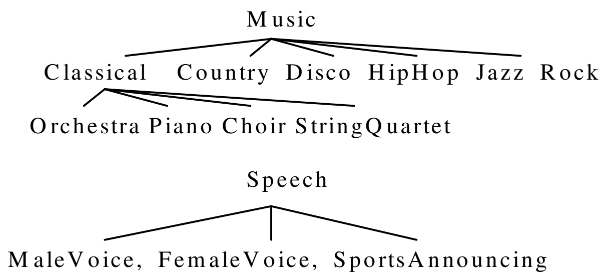

Fig. III Genre Classification Hierarchy

Table 1. Classification accuracy percentage results

	MusicSpeech	**Genres**	**Voices**	**Classical**
Random	50	16	33	25
Gaussian	86	62	74	76

Table 1. summarizes the classification results as pecentages of classification accuracy. In all cases the results are significantly better than random classification. These classification results are calculated using a 10-fold evaluation strategy where the evaluation data set is randomly partitioned so that 10% is used for testing and 90% for training. The process is iterated with different random partitions and the results are averaged (in the evaluation of Table.1 one hundred iterations where used).

Table 2. shows more detailed information about the genre classifier performance in the form of a confusion matrix. The columns correspond to the actual genre and the rows to the predicted genre. For example the cell of row 2, column 1 with value 0.01 means that 1 percent of the Classical music (column 1) was wrongly classified as Country music (row 2). The percentages of correct classifications lie in the diagonal of the confusion matrix. The best predicted genres are classical and hiphop while the worst predicted are jazz and rock. This is due to the fact that the jazz and rock are very broad categories and their boundaries are more fuzzy than classical or hiphop.

Table 3. shows more detailed information about the classical music classifier performance in the form of a confusion matrix..

	classic	*country*	*Disco*	*Hiphop*	*jazz*	*Rock*
classic	86	2	0	4	18	1
country	1	57	5	1	12	13
disco	0	6	55	4	0	5
Hiphop	0	15	28	90	4	18
Jazz	7	1	0	0	.37	12
Rock	6	19	11	0	27	48

Table 2. Genre classification confusion matrix

	choral	*orchestral*	*Piano*	*string 4tet*
choral	99	10	16	12
orchestral	0	53	2	5
piano	1	20	75	3
string 4tet	0	17	7	80

Table 2. Classical music classification confusion matrix

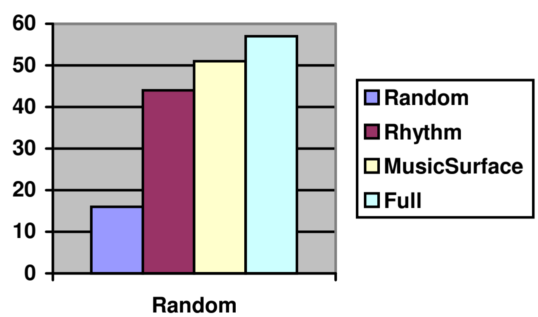

Fig. IV Relative feature set importance

Figure IV shows the relative importance of the "musical surface" and "rhythm" feature sets for the automatic genre classification. As expected both feature sets perform better than random and their combination improves the classification accuracy. The genre labeling was based on the artist or the compact disk that contained the excerpt. In some cases this resulted in outliers that are one of the sources of prediction error. For example the Rock collection contains songs by Sting that are more close to Jazz than Rock even for a human listener. Similarly the Jazz collection contains songs with string accompaniment and no rhythm section that sound like Classical music. It is likely that replacing these outliers with more characteristic pieces would improve the genre classification results.

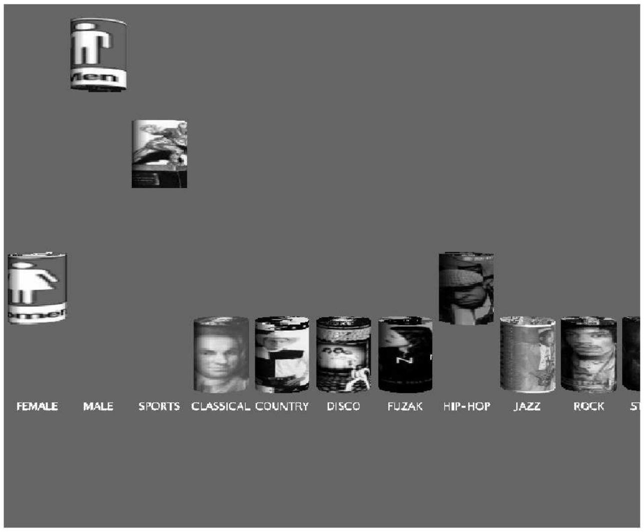

Fig. IV GenreGram

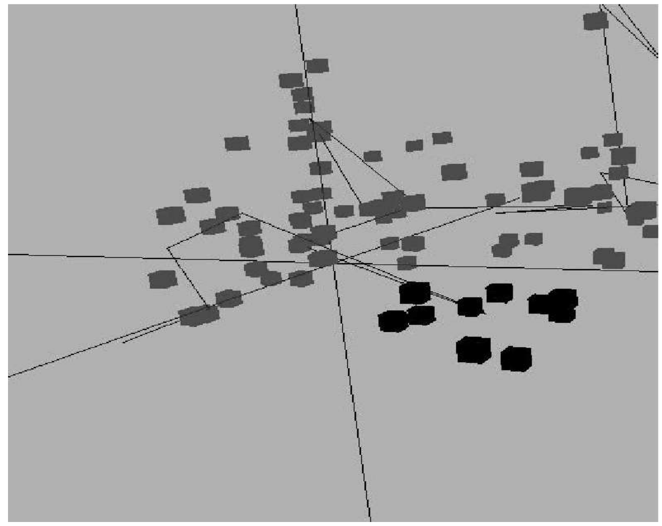

Fig. V GenreSpace

5. USER INTERFACES

Two new graphical user interfaces for browsing and interacting with collections of audio signals have been developed (Figure IV,V) . They are based on the extracted feature vectors and the automatic genre classification results.

- **GenreGram** is a dynamic real-time audio display for showing automatic genre classification results. Each genre is represented as a cylinder that moves up and down in real time based on a classification confidence measure ranging from 0.0 to 1.0. Each cylinder is texture-mapped with a representative image of each category. In addition to being a nice demonstration of automatic real time audio classification, the *GenreGram* gives valuable feedback both to the user and the algorithm designer. Different classification decisions and their relative strengths are combined visually, revealing correlations and classification patterns. Since the boundaries between musical genres are fuzzy, a display like this is more informative than a single classification decision. For example, most of the time a rap song will trigger *Male Voice, Sports Announcing* and *HipHop*. This exact case is shown in Figure IV.

- **GenreSpace** is a tool for visualizing large sound collections for browsing. Each audio file is represented a single point in a 3D space. Principal Component Analysis (PCA) [23] is used to reduce the dimensionality of the feature vector representing the file to the 3-dimensional feature vector corresponding to the point coordinates. Coloring of the points is based on the automatic genre classification. The user can zoom, rotate and scale the space to interact with the data. The *GenreSpace* also represents the relative similarity within genres by the distance between points. A principal curve [24] can be used to move sequentially through the points in a way that preserves the local clustering information.

6. FUTURE WORK

An obvious direction for future research is to expand the genre hierarchy both in width and depth. The combination of segmentation [25] with automatic genre classification could provide a way to browse audio to locate regions of interest. Another interesting direction is the combination of the graphical user interfaces described with automatic similarity retrieval that takes into account the automatic genre classification. In its current form the beat analysis algorithm can not be performed in real time as it needs to collect information from the whole signal. A real time version of beat analysis is planned for the future. It is our belief that more rhythmic information can be extracted from audio signals and we plan to investigate the ability of the beat analysis to detect rhythmic structure in synthetic stimuli.

7. SUMMARY

A feature set for representing music surface and rhythm information was proposed and used to build automatic genre classification algorithms. The performance of the proposed data set was evaluated by training statistical pattern recognition classifiers on real-world data sets. Two new graphical user interfaces based on the extracted feature set and the automatic genre classification were developed.

The software used for this paper is available as part of *MARSYAS* [26] a software framework for rapid development of computer audition application written in C++ and JAVA. It is available as free software under the Gnu Public License (GPL) at:

http://www.cs.princeton.edu/~gtzan/marsyas.html

8. ACKNOWLEDGEMENTS

This work was funded under NSF grant 9984087 and from gifts from Intel and Arial Foundation. Douglas Turnbull helped with the implementation of the GenreGram. The authors would also like to thank the anonymous reviewers for their valuable feedback.

9. REFERENCES

[1] Perrot, D., and Gjerdigen, R.O. Scanning the dial: An exploration of factors in the identification of musical style. In Proceedings of the 1999 Society for Music Perception and Cognition pp.88(abstract)

[2] Martin, K.,D., Scheirer, E.D., Vercoe, B., L. Musical content analysis through models of audition. In Proceedings of the 1998 ACM Multimedia Workshop on Content-Based Processing of Music.

[3] Foote, J. An overview of audio information retrieval. Multimedia Systems 1999. 7(1), 42-51.

[4] Scheirer, E. D. and Slaney, M. Construction and evaluation of a robust multifeature speech/music discriminator. In Proceedings of the 1997 International Conference on Acoustics, Speech, and Signal Processing, 1331-1334.

[5] Wold, E., Blum, T., Keislar, D., and Wheaton, J. Content – based classification, search and retrieval of audio. IEEE Multimedia, 1996 3 (2)

[6] Foote, J., Content-based retrieval of music and audio. In Multimedia Storage and Archiving Systems II, 1997 138-147

[7] Martin, K. Sound-Source Recognition: A theory and computational model. PhD thesis, MIT Media Lab. http://sound.media.mit.edu/~kdm

[8] Rossignol, S et al. Feature extraction and temporal segmentation of acoustic signals. In Proceedings of International Computer Music Conference (ICMC), 1998.

[9] Dubnov, S., Tishby, N., and Cohen, D. Polyspectra as measures of sound and texture. Journal of New Music Research, vol. 26 1997.

[10] Scheirer, E. Music Listening Systems. Phd thesis., MIT Media Lab: http://sound.media.mit.edu/~eds

[11] Welsh, M., Borisov, N., Hill, J., von Behren, R., and Woo, A. Querying large collections of music for similarity. Technical Report UCB/CSD00-1096, U.C Berkeley, Computer Science Division, 1999.

[12] Scheirer, E. Tempo and beat analysis of acoustic musical signals. Journal of the Acoustical Society of America 103(1):588-601.

[13] Goto, M. and Muraoka, Y. Music understanding at the beat level: real time beat tracking for audio signals. In D.F Rosenthal and H. Okuno (ed.), Readings in Computational Auditory Scene Analysis 156-176.

[14] Gouyon, F., Pachet, F. and Delerue, O. On the use of zero-crossing rate for an application of classification of percussive sounds. Proceedings of the COST G-6 conference on Digital Audio Effects (DAFX-00), Verona, Italy, 2000.

[15] Pachet, F., Cazaly, D. “A classification of musical genre”, Content-Based Multimedia Information Access (RIA) Conference, Paris, March 2000.

[16] Oppenheim, A. and Schafer, R. Discrete-Time Signal Processing. Prentice Hall. Edgewood Cliffs, NJ. 1989.

[17] Mallat, S, G. A theory for multiresolution signal decomposition: The Wavelet representation. IEEE Transactions on Pattern Analysis and Machine Intelligence,1989, 11, 674-693.

[18] Mallat, S,G. A wavelet tour of signal processing. Academic Press 1999.

[19] Daubechies, I. Orthonormal bases of compactly supported wavelets. Communications on Pure and Applied Math.1988. vol.41, 909-996.

[20] Duda, R. and Hart, P. Pattern classification and scene analysis. John Willey & Sons. 1973.

[21] Hunt, M., Lennig, M., and Mermelstein, P. Experiments in syllable-based recognition of continuous speech. In Proceedings of International Conference on Acoustics, Speech and Signal Processing, 1996, 880-883.

[22] Logan, B. Mel Frequency Cepstral Coefficients for music modeling. Read at the first International Symposium on Music Information Retrieval.. http://ciir.cs.umass.edu/music2000

[23] Jollife, L. Principal component analysis. Spring Verlag, 1986.

[24] Herman, T, Meinicke, P., and Ritter, H. Principal curve sonification. In Proceedings of International Conference on Auditory Display. 2000.

[25] Tzanetakis, G. and Cook, P. Mutlifeature audio segmentation for browsing and annotation. In Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. 1999.

[26] Tzanetakis, G. and Cook, P. MARSYAS: a framework for audio analysis. Organised Sound 2000. 4(3)